Title

Applications of unsupervised machine learning in classification: Detecting optimal clustering method for baby cry

Author

Lyu, Hongze

Publication Date

2025

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Applications of unsupervised machine learning in classification: Detecting optimal clustering method for baby cry

A thesis submitted in partial satisfaction

of the requirements for the degree

Master of Applied Statistics and Data Science

by

Hongze Lyu

2025

ABSTRACT OF THE THESIS

Applications of unsupervised machine learning in classification:
Detecting optimal clustering method for baby cry

by

Hongze Lyu
Master of Applied Statistics and Data Science
University of California, Los Angeles, 2025
Professor Yingnian Wu, Chair

Clustering baby cry sounds can provide valuable insights into infant needs and potential health conditions. However, selecting the optimal clustering method for such an acoustic dataset remains a challenge. This study explores various unsupervised clustering techniques to determine the most effective approach for grouping baby cries based on their acoustic features. We evaluate methods including K-means, hierarchical clustering, DBSCAN, spectral clustering and Self-Organizing Maps (SOM), analyzing their performance in terms of cluster separation and consistency. A key focus is on assessing clustering validity using internal metrics such as the Silhouette Score and Davies-Bouldin Index. Our findings indicate that certain methods, particularly SOM combined with K-means, provide well-separated clusters, though challenges remain in ensuring robustness across different cry patterns. The results contribute to the broader understanding of infant vocalization analysis and offer a foundation for future studies in automated baby cry classification.

The thesis of Hongze Lyu is approved.

Ariana Anderson

Maryam Mahtash Esfandiari

Yingnian Wu, Committee Chair

University of California, Los Angeles

2025

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1.1 Introduction of the Thesis

It's been a long story that parents are so eager to figure out why their babies are crying. On one hand, they want to solve the uncomfort of their kids. On the other hand, they also in demand for a quiet and peace life. However, it's really hard to determine what causes the cry by human's ear. According to [Bum], there are 11 reasons behind baby's crying and parents can only solve the problem if they want to calm the baby down. It is much harder than it sounds.

With a strong desire to solve this problem, my thesis advisor, Dr. Ariana Anderson, is dedicated to build up a machine learning model for parents so that they can record the crying sound of baby and receive a potential reason causing their baby's crying through a phone app. In order to achieve this goal, one of the crucial step is that we should define how many types of reasons are there behind baby cry. Is it as much as 11 like [Bum] has said? Or is there better classification ways? In previous studies, Dr. Anderson and her team studied and assumed that there should be 5 clusters. However, as they are building up predictive models, they met a serious overlapping problem among 5 clusters. As a result, Dr. Anderson invites me to use unsupervised learning models to detect if there are better ways to separate clusters. In this case, I would treat the number of cluster as a target unknown parameter. Based on Dr. Anderson's study, our best guess is that the number of clusters should be equal to or smaller than 5. Also, not only the number of clusters is unknown, if we detect there are fewer clusters than expected, current true label will no longer be valid. That's why I'll focus on using unsupervised learning models to solve this problem.

1.2 Background: The ChatterBaby's Project

[Cha]is a research initiative supported by UCLA Health, led by Dr. Ariana Anderson. The organization focuses on understanding the science behind the crying of infants using statistical models and machine learning techniques. One of the primary goals of ChatterBaby is to help parents identify the reasons behind their babies' cries, such as hunger, pain, or tiredness, by analyzing the acoustic features of the cries. By leveraging a well-organized dataset collected and cleaned by the ChatterBaby team, this project aims to develop tools that can provide real-time insights into infant communication, ultimately reducing parental stress and improving caregiving.

CHAPTER 2

Data Exploration

2.1 Basic Information of the Dataset

According to Dr. Anderson:

The five most common types of cries labeled by parents were first identified in the ChatterBaby database. All audio samples underwent pre-processing and screening with ChatterBaby's cry detection and identification algorithms to ensure the recordings were exclusively baby cries, not other sounds.

1250 samples were acquired from the five most common cry labels as part of the ChatterBaby infant cry study, providing 250 representative cries per category. The study population consisted of 1250 unique infants recorded through the app. Gender and age were statistically balanced: 50.15% female and babies were on average 146 days old for the males and 161 days old for the females on the recording date.

Each audio clip was 5 seconds long and recorded as 16-bit PCM WAV files. Audio files were processed using the OpenSmile GeMaps acoustic feature set. The OpenSMILE GeMAPS (Geneva Minimalistic Acoustic Parameter Set) extracts 62 acoustic features in its Extended (eGeMAPS) version and 18 features in its Minimal (GeMAPS) version. This provides a low-dimensional feature space describing the acoustic characteristics of the cries. Our previous work has shown that using higher-dimensional feature spaces does not drastically improve the ability of classification algorithms but may increase the risk of overfitting. The GeMAPS set includes fundamental frequency (F0), jitter and shimmer measures, harmonic-to-noise ratio (HNR), spectral balance features such as formants and energy distribution across frequency bands, and various voice quality measures. This feature set is widely used in speech analysis and provides

a low-dimensional feature space describing the acoustic characteristics of the cries.

Here, we define "category" as the parent label consisting of tired, pain, hungry, fussy, and diaper change. We define a cluster as a group of sounds with similar acoustic features created by clustering algorithms agnostic to the parent labels.

2.2 Exploratory Data Analysis (EDA)

To start my analysis, I think it will be beneficial to my study if I can have a clear idea of how these features vary and how they will contribute to the clustering process. So I did some exploratory data analysis to those features. What I did first was plot a decision tree to have an arbitrary view of the importance of the features. From Figure 1 below, we can see that *MeanUnvoicedSegmentLength*, *loudness_sma3_percentile50.0*, *slopV500.1500_sma3nz_amean*, and *loudness_sma3_meanRisingSlop* are several features that have the most important influence in separating the clusters. To get a better understanding, we can take some further look at these four features. To make the analysis more concise, I standardized all the features before analyzing them.

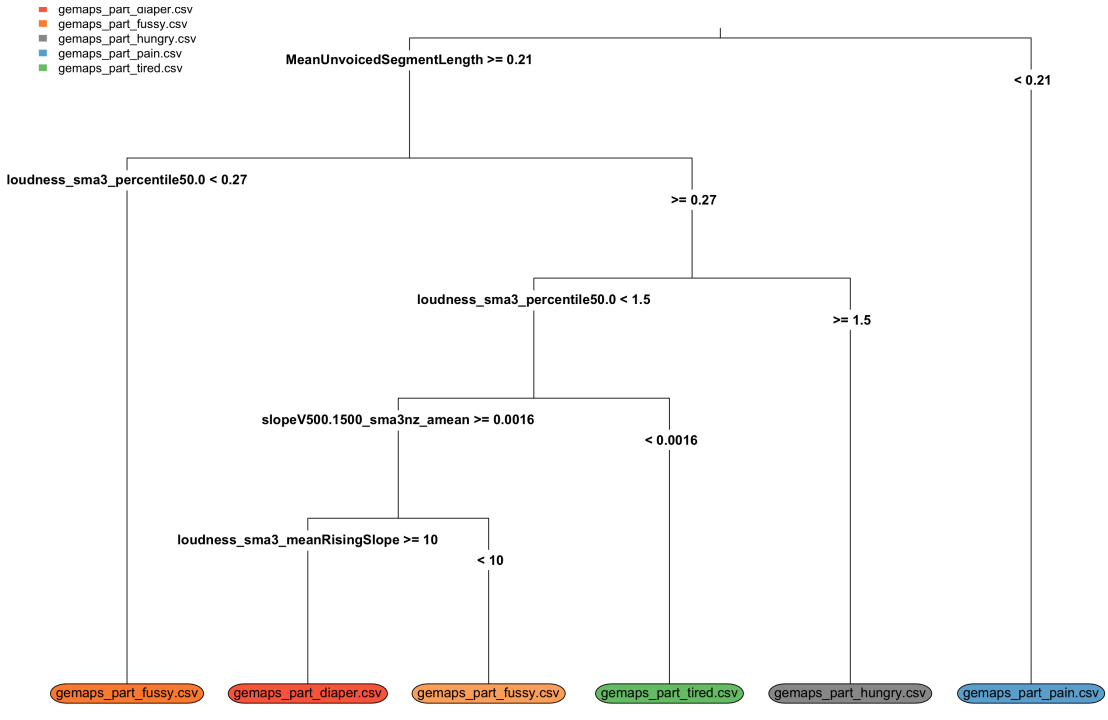


Figure 2.1: Decision Tree Visualization for first several key features

2.2.1 Mean Unvoiced Segment Length

In Figure 1, the first feature that makes an impact to separate pain group with other groups is the Mean Unvoiced Segment Length, which directly differentiates the data in a pain group. Then I plot the histogram of this feature. From Figure 2, we can see that this feature shows a strong right-skewed pattern and the data in the pain cluster were the most skewed, which is associated with the result of the decision tree graph.

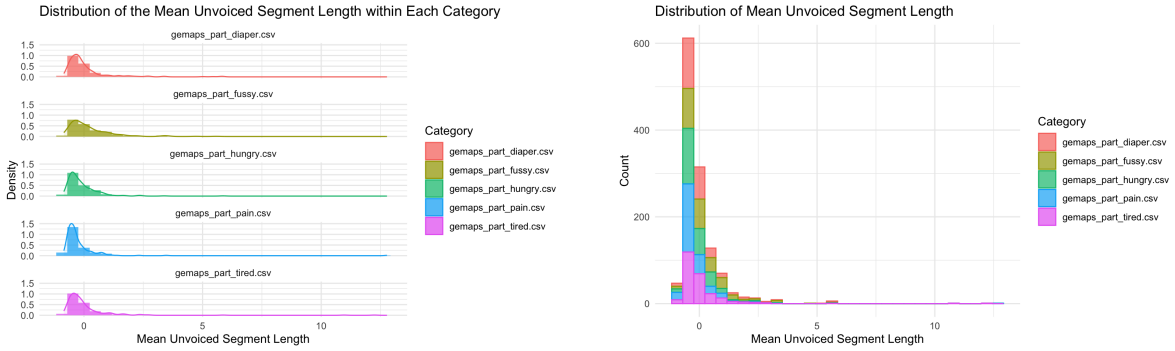


Figure 2.2: Histogram of MUSL

2.2.2 Loudness sma3 percentile 50.0

In Figure 1, we can see that the second and third decisions are all made by the feature *loudness_sma3_percentile50.0*. In Figure 3, we can see that although all the features are right-skewed. The distribution for pain group and hungry group tends to be less skewed and flatter compared to other three groups.

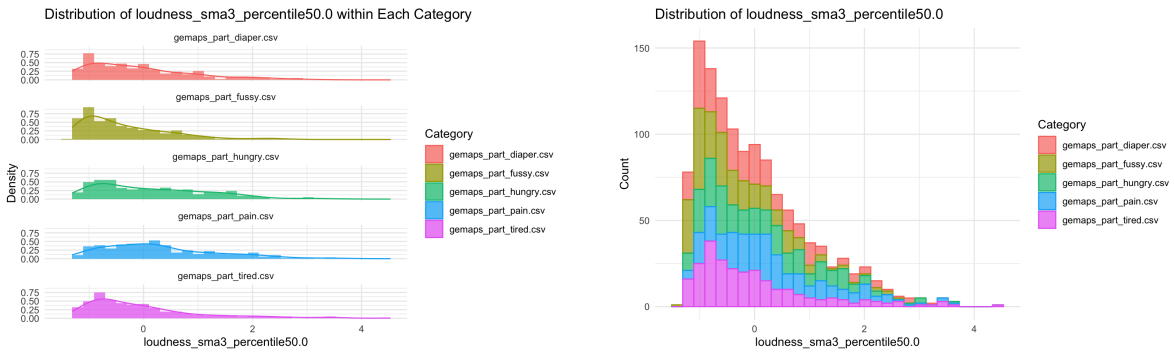


Figure 2.3: Histogram of loudness sma3 percentile 50.0

2.2.3 SlopeV500.1500 sma3nz amean

In Figure 1, we can see that the fourth decision is made by the feature *slopV500.1500_sma3nz_amean* to differentiate tired group. In Figure 4, we observe two kinds of distributions. Diaper change and Fussy group shows a unimodel distribution while Pain and Tired groups shows a strong bimodel distribution pattern, leaving hungry group be somewhere in the middle. Since the decision tree has separated pain group in the beginning, Figure 3 supports the result of the decision tree model.

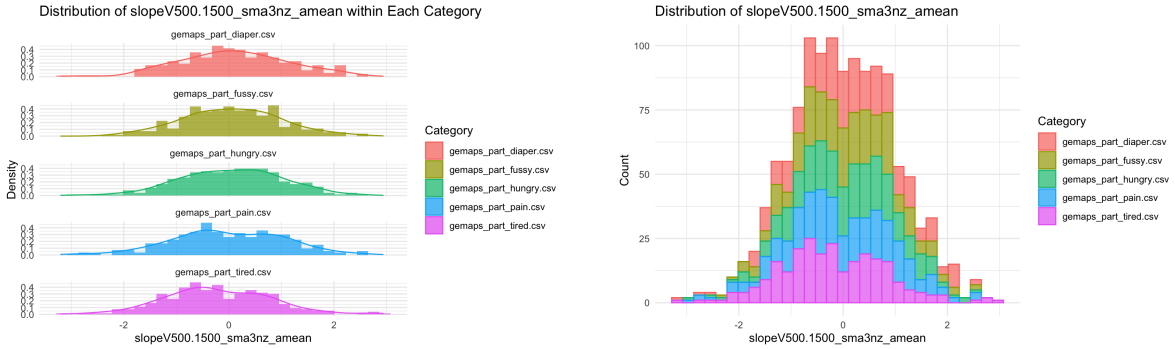


Figure 2.4: Histogram of SlopeV500.1500 sma3nz amean

2.2.4 Loudness Sma3 MeanRisingSlope

In Figure 1, we can see that the fifth decision is made by feature *loudness_sma3_meanRisingSlop* to differentiate fussy and diaper change group. However, unlike previous features' histograms that strongly support the decision, no significant difference of distribution is observed between fussy and diaper change group, which encourages me to pay attention to the potential overlapping problem within these two groups.

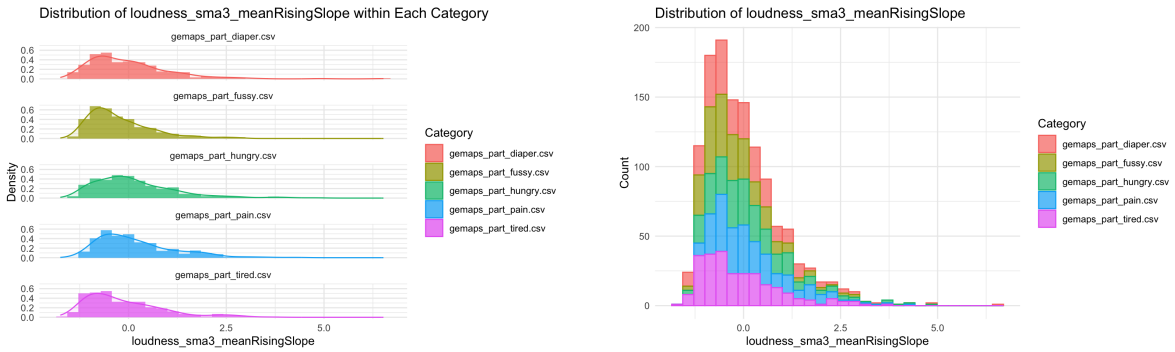


Figure 2.5: Histogram of Loudness Sma3 MeanRisingSlope

2.2.5 Conclusion of EDA

By comparing the histograms of each key features detected by decision tree model with each decisions, I find that fussy group and diaper change groups show the strongest possibility of overlapping problem. But overall, all the five clusters have similar distribution patterns with minor variation, which supports our assumption that the clusters have overlapping issue and it will be better if we can make a clustering set with higher separation rate.

2.3 Data Visualization

To begin with, it would be helpful for us to have a better understanding of the "true labels" based on our current assumption that there are five clusters addressing baby cry (diaper change, fussy, hungry, pain and tired). Here, I used two methods to do visualization, and each has its own advantages. First, I tried to use UMAP to visualize the geographic information of the original features. One of the biggest advantages of the UMAP method is that it will remain the geographic information of the data so that we can have a direct understanding of our data. Then I used TSNE to visualize the standardized features to have a better understanding of the cluster information.

2.3.1 UMAP Visualization of the original data

From Figure 6, we find that the 2d projection of our data is shaped like a crescent. However, all the data points from different clusters are mixed together, and the individual shape of each cluster looks identical. So we hope to observe more inter-cluster information from the T-SNE visualization below.

2.3.2 T-SNE Visualization of the scaled data

Since the geographic meaning of the data doesn't provide us much useful information, I standardized the data in T-SNE visualizaion and also in the following analysis part to make our result better. There's limited information we can get from Figure 7, because

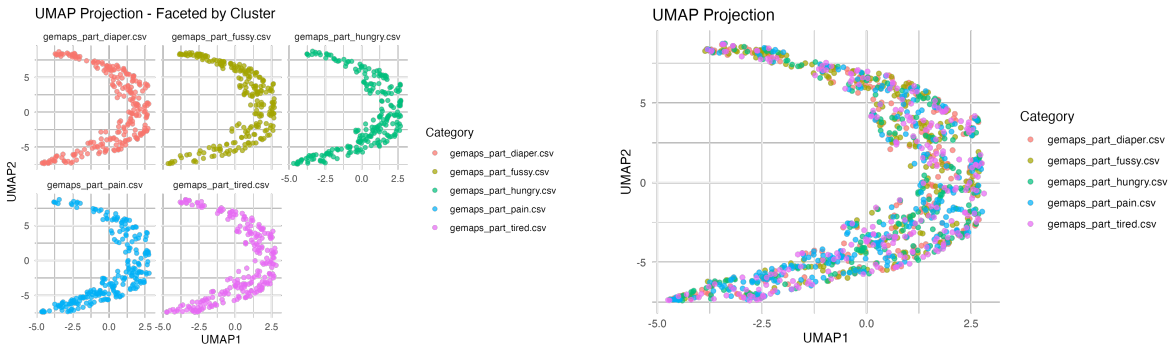


Figure 2.6: UMAP visualization

the data are still too mixed to be seen for each cluster.

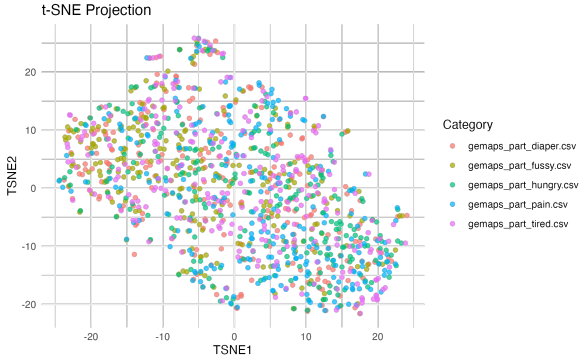


Figure 2.7: T-SNE visualization

But if we take a closer look at the facet figure, there are some patterns that might be valuable. One pattern that I notice is that the density of data is different between clusters. For example, fussy cluster is more dense at (-20,0) and less dense around (20,-20), while cluster hungry and cluster pain are exactly the opposite. Cluster diaper and cluster tired seems to be evenly distributed. These patterns might also be worth paying attention in the following analysis.

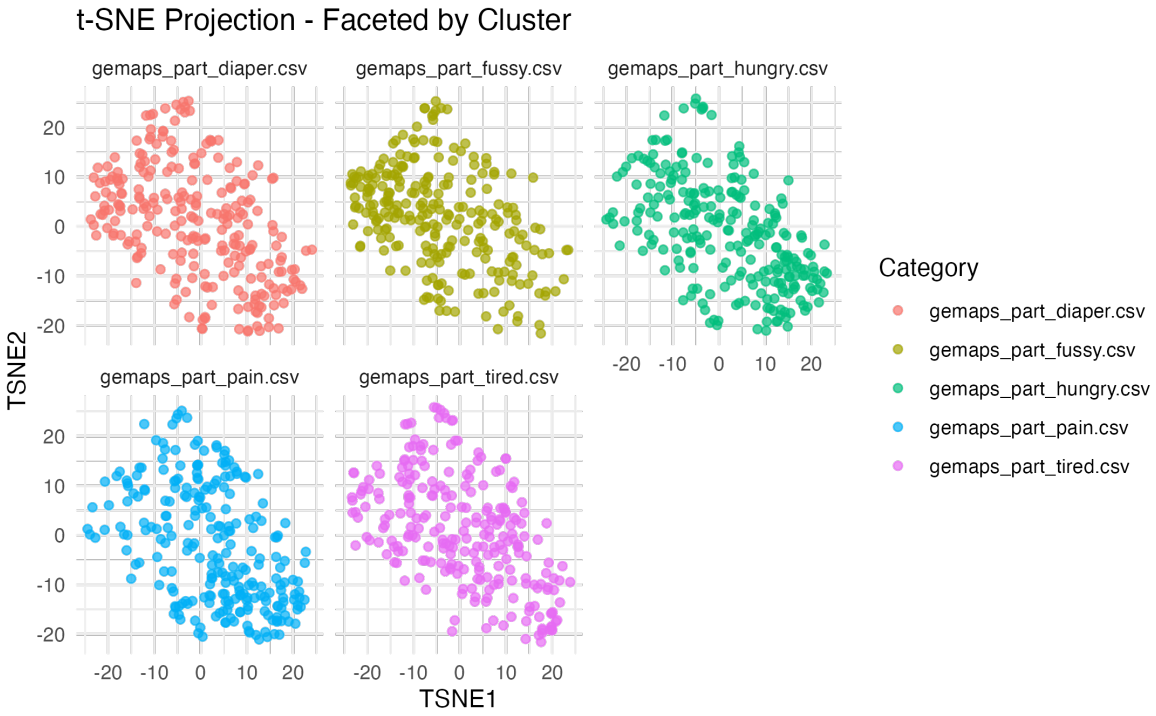


Figure 2.8: facet T-SNE visualization

CHAPTER 3

Comparison of Unsupervised Clustering Methods

In this Chapter, I'll use different unsupervised machine learning models to solve the same clustering problem to find out the optimal clustering method.

3.1 Introduction of methodology

Since our goal is to find the optimal number of clusters for the data, it is important to set evaluation criteria. I choose to use silhouette score and Davies-Bouldin Index, which are two commonly used indices describing the quality of the clusters. I also calculated the V-measure and

3.1.1 Silhouette Score

The **Silhouette Score** measures how well-separated clusters are by quantifying the cohesion (how close data points in the same cluster are) and separation (how distinct different clusters are). For a data point i, the Silhouette Score is calculated as:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

Where:

- $a(i)$: Average distance between i and all other points in the same cluster (intra-cluster distance).
- $b(i)$: Average distance between i and points in the nearest different cluster (inter-cluster distance).

The score ranges from:

- +1: Points are well matched to their own cluster and poorly matched to others.
- 0: Points are on or near the cluster boundary.
- −1: Points are assigned to the wrong cluster.

The overall Silhouette Score is the mean score of all data points. Higher scores indicate better-defined clusters.

3.1.2 Davies-Bouldin (DB) Index

The **Davies-Bouldin Index** evaluates clustering quality by comparing the ratio of within-cluster scatter to the between-cluster separation. It is given by:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{s_i + s_j}{d_{ij}} \right)$$

Where:

- s_i: Average distance between points in cluster i and the centroid of cluster i (cluster scatter).
- d_{ij}: Distance between the centroids of clusters i and j.

Key characteristics:

- Lower values indicate better clustering (compact and well-separated clusters).
- The DB Index is less computationally intensive compared to the Silhouette Score but assumes spherical clusters.

3.1.3 V-measure

The **V-measure** is an entropy-based metric that evaluates clustering by measuring both homogeneity (all samples in a cluster belong to the same class) and completeness

(all samples of a class are assigned to the same cluster). It is defined as:

$$V = \frac{(1+\beta) \cdot H \cdot C}{(\beta \cdot H) + C}$$

Where:

- H: Homogeneity score.
- C: Completeness score.
- β: Weighting factor (default is $\beta = 1$, giving equal weight to H and C).

Interpretation:

- Values range from 0 to 1.
- A higher score indicates clusters that are both homogeneous and complete.

3.1.4 Chi-Square Test

The **Chi-Square Test** evaluates the independence between two categorical variables, often used to assess the relationship between cluster assignments and true labels in supervised or semi-supervised clustering. The test statistic is:

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Where:

- O_{ij}: Observed frequency in cell (i, j).
- E_{ij}: Expected frequency under the null hypothesis.

Interpretation:

- A higher χ^2 value indicates a stronger association between the variables.
- Often accompanied by a p-value to determine statistical significance.

3.1.5 Comparison of Metrics

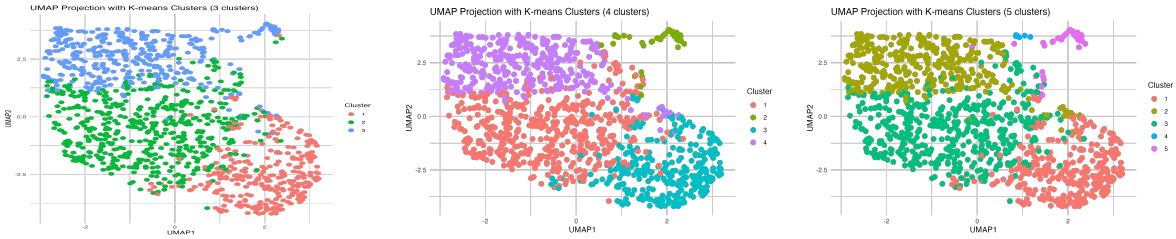
Metric	Range	Optimal Value	Description
Silhouette Score	$[-1, 1]$	Close to 1	Higher values indicate well-separated clusters.
Davies-Bouldin Index	$[0, \infty)$	Close to 0	Lower values indicate compact, well-separated clusters.
V-measure	$[0, 1]$	Close to 1	Higher values indicate clusters that are homogeneous and complete.
Chi-Square Test	$[0, \infty)$	Depends on p-value	Tests the statistical relationship between clusters and true labels.

Table 3.1: Comparison of Clustering Evaluation Metrics

3.2 K-means

As a start, I use the simplest mode: K-means clustering. K-means clustering is an unsupervised machine learning algorithm that groups data points into a pre-defined number of clusters ("k") based on their similarity. Since we have 5 overlapping clusters, I set k = 3,4,5 and observe the result of cluster.

3.2.1 Visualization of K-means



(a) 3 Clusters (b) 4 Clusters (c) 5 Clusters

Figure 3.1: UMAP visualization of K-means clustering with varying cluster numbers.

From Figure 9, we can see that the K-means clustering results with $k = 3$, $k = 4$, and $k = 5$ don't differentiate a lot. According to the graph, having one or two more clusters from 3 clusters is adding some clusters addressing the minor outliers for the

data.

3.2.2 Summary table of K-means result

From table 1, we find the optimal number of clusters gained by K-means clustering method is 5, with a silhouette score of 0.158 and DBI of 1.656. However, the result of K-means clustering doesn't vary a lot and it is not so satisfying. All the indices indicate that there is strong overlapping problems among the clusters, which encourages us to keep trying other methods.

n clusters	Silhouette Score	Davies-Bouldin Index	V-measure	Chi Square p_value
3	0.1386	2.1674	0.0188	$1.39 \cdot 10^{-10}$
4	0.1513	1.9294	0.0183	$2.98 \cdot 10^{-9}$
5	0.1582	1.6559	0.0196	$1.75 \cdot 10^{-8}$

Table 3.2: K-means clustering result

3.3 DBSCAN

DBSCAN model stands for Density-Based Spatial Clustering of Applications with Noise, which is a popular density-based clustering algorithm that identifies clusters as regions of high density separated by regions of low density. This model requires some manual control of the hyper-parameters.

From the plot of k-distance (figure 10), we can see there is an elbow turn in 5, which shows the most proper value of epsilon for DBSCAN in this case is around 5. Then we can iterate on different hyperparameters to find the best clustering method. Here I used eps=5 and minPts=30

3.3.1 DBSCAN visualization

According to DBSCAN clustering result, most of the data points should be in one cluster, leaving the rest few in an outlier cluster.This figure gives us another message that potentially kids cannot differentiate the cause of crying and they are just reacting to their unconfortableness. However, since our ultimate goal is to create a AI model

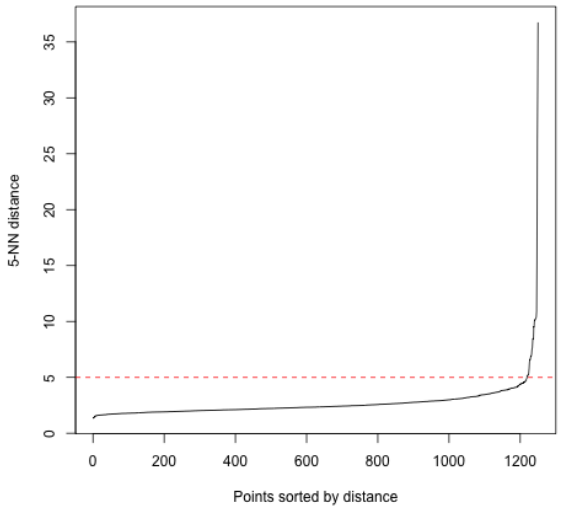


Figure 3.2: K-distance figure

helping parents detect the reason behind kids' crying, this pattern is not a direction that we would go further.

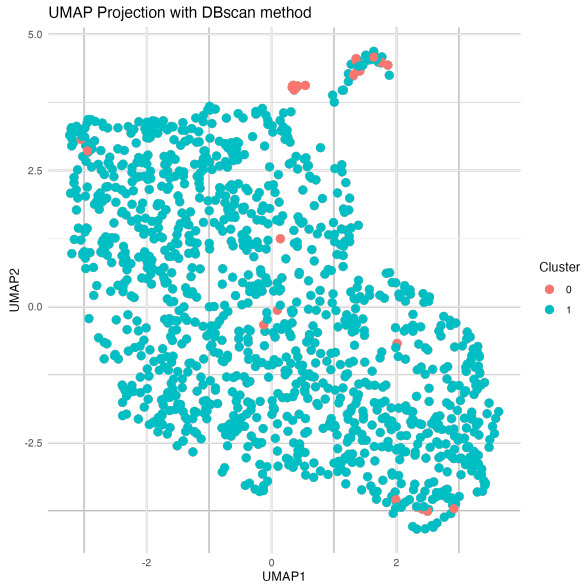


Figure 3.3: UMAP visualization of DBSCAN model

3.3.2 Summary of DBSCAN model

Although it can generate a clustering method with a high Silhouette score of 0.59887064672121, we cannot gain enough practical value from this clustering model. This process is re-

peated many times using different hyperparameters to reduce the cause of randomness. However, all the clustering result shows DBscan model shows similar result as Figure 11, which has a tendency to create a major cluster with a small minor cluster. So it is not suitable for our study.

3.4 Hierarchical clustering

Hierarchical clustering is an unsupervised machine learning technique used to group data into clusters based on their similarity. It doesn't require us to set the number of clusters for it. Instead, it will produce a hierarchy of clusters that can be visualized using a dendrogram, like the decesion tree we made in the EDA section.

3.4.1 Dengrogram of hierarchical clustering

Since our previous assumption is to have 5 clusters, I set $k = 5$ in the dendrogram to have a better understanding of our data. Similar to what we get from K-means clustering method, the dendrogram shows 3 main clusters (in color red, green and blue) with 2 minor clusters (in color yellow and purple), which encourages us to see if its possible to cut the number of clusters to 3.

3.4.2 Summary table of hierarchical clustering

clusters	Silhouette Score	Davies Bouldin Index	V measure
3	0.1736	2.0825	0.0138
4	0.1007	2.3299	0.0133
5	0.1031	1.7230	0.0142

Table 3.3: K-means clustering result

We can see that the result of Hierarchical clustering suggests that we should cut the number of clusters to 3 with a silhouette score of 0.1736 with a BDI of 2.0825. Still, this value is not satisfying enough.

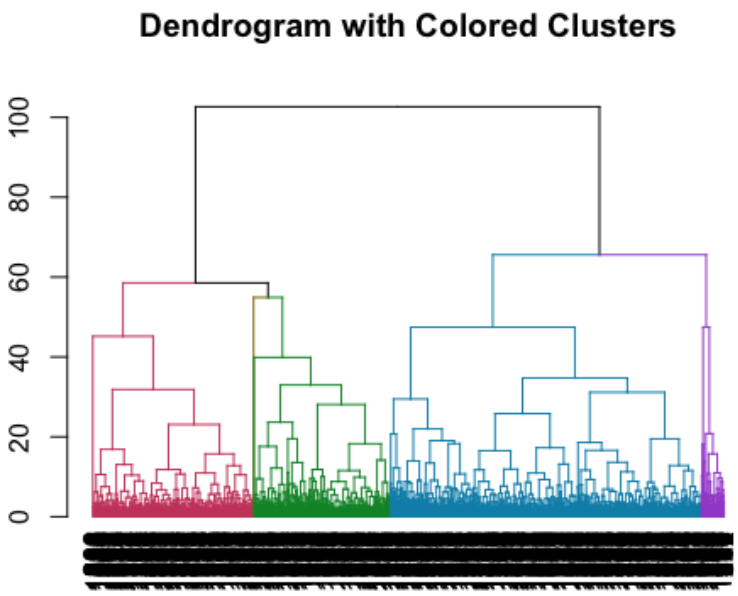


Figure 3.4: Dendrogram plot with k = 5

3.4.3 Visualizaion of Hierarchical clustering with k = 3

From figure 13, we can see the clustering result of hierarchical clustering with $k = 3$ contains two main clusters and one minor clusters(purple part of the dendrogram).

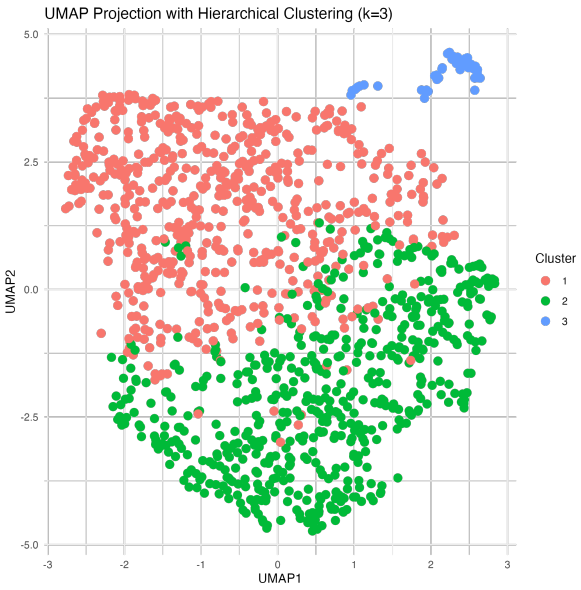


Figure 3.5: UMAP visualization of hierarchical clustering with k = 3

3.5 Self-Organizing Maps (SOM)

SOM is a dimension reduction method normally being used in clustering problems. We can also try SOM followed by K-means clustering. Comparing to directly using K-means clustering method, SOM first reduce the dimension of the data, which will not only help avoid the curse of dimensionality, but also preserve the topological relationship among the data points. Since our data is most likely non-linear, SOM can help K-means method to produce a better result.

3.5.1 Summary and visualization of SOM result

The result of SOM is the most satisfying till now. The optimal output occurs when there are three clusters. It reaches a silhouette score of 0.2566 and a DBI of 1.4391, which is the best result we can get so far.

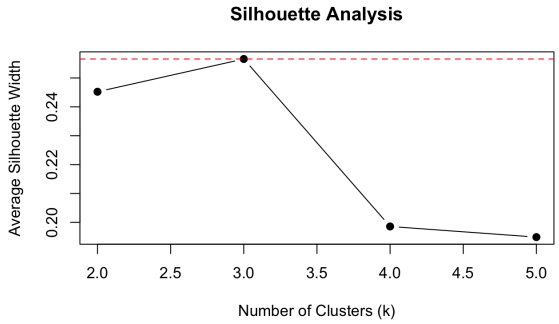


Figure 3.6: Silhouette score of SOM

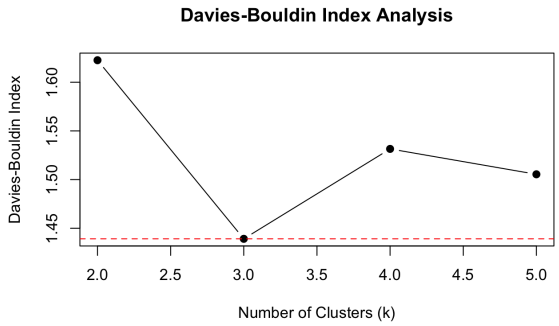


Figure 3.7: DBI of SOM

From figure 16 (UMAP visualization of SOM with 3 clusters), we can observe that

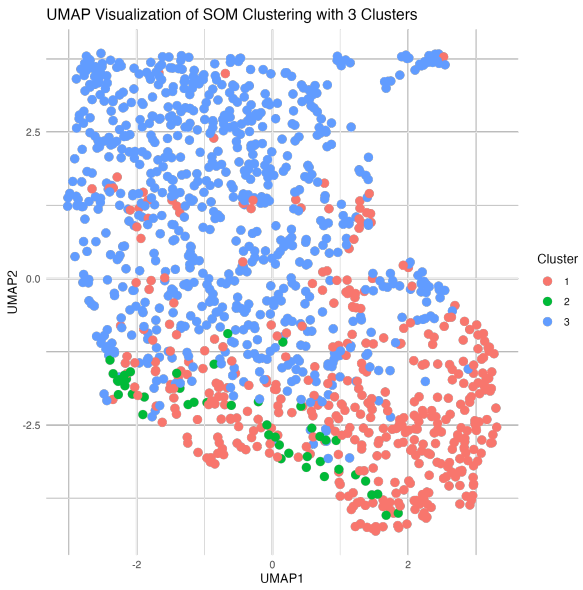


Figure 3.8: UMAP of SOM with 3 clusters

it shows a similar pattern of what we get from hierarchical clustering method where there are two main clusters with one minor cluster. But the minor cluster contains different part of the data and SOM result is more mixed geomatrically comparing to hierarchical clustering result.

3.6 Spectral clustering

As suggested by Professor Wu, I also tried spectral clustering method. Spectral clustering is a relatively new clustering method that fully applies the graphical information of data to separate clusters.It is particularly effective for non-linear and complex-shaped clusters that cannot be captured by traditional clustering methods.

3.6.1 Summary and visualization of Spectral Clustering

clusters	Silhouette Score	Davies-Bouldin Index
2	0.9644	0.3933
3	0.9660	0.4415
4	0.9601	0.3129
5	0.9418	0.3370

Table 3.4: K-means clustering result

Although this method generates a significantly good evaluation score, the plot of this method shows it also treats most data in one cluster, similar to the situation of DBSCAN model. For the same reason, this is not the direction we would go in.

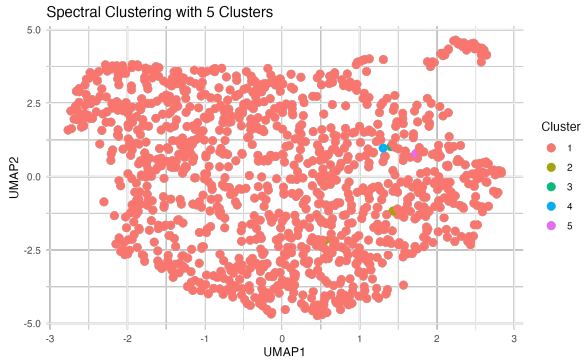


Figure 3.9: Spectral clustering with k = 5

3.7 Comparative Analysis of Clustering Methods

Among all these methods, we can see there are two types of results. The first type includes DBSCAN model and Spectral clustering model which tends to treat almost all data points in one cluster, leaving other clusters containing some outliers from the main. Although they can produce a result with significantly high evaluation score, the clustering result shows no pratical value to our study. The second type includes the result of K-means clustering, Hierarchical cluster and Self-organizing map results, which properly separates the clusters. Their evaluation scores varies a lot but the plots all indicates that having two to three clusters might have a better performace than having 5 clusters. In the end, SOM shows the significantly best performance among three methods, which encourages us to go deeper in this method and see if we can improve it to a higher level.

CHAPTER 4

Optimizing SOM for Improved Clustering

From the previous chapters, we find SOM model shows the best performance and might have the biggest potential in our study. As a result, we would like to try improving the performance of the model.

4.1 Tuning parameters

In general, there are two important parameter affecting the performance of SOM model. First is the grid size, which is strongly related to number of clusters the model can form. With larger grid size, it will include more details when separating clusters which can improve the accuracy of model but might result in overfitting. The second parameter is learning rate, which controls the convergence speed.

In the beginning, since our assumption sets the potential cluster number around 5 or less, I would try smaller grid sizes rather than regular sizes for higher accuracy. To prove my thought, I tried grid size with 5,10,15 and 20. The result indicates larger grid size won't contribute to our model.

Silhouette Scores for Different Grid Sizes and Learning Rates

	LR 0.01	LR 0.05	LR 0.1	LR 0.2
Grid Size 5	0.5057961	0.6163758	0.6430077	0.6713799
Grid Size 10	0.2462380	0.2297986	0.2190585	0.2031029
Grid Size 15	0.2502272	0.2522047	0.2426473	0.2356440
Grid Size 20	0.2645220	0.2536954	0.2520840	0.2489393

Figure 4.1: Grid Search Matrix of SOM parameter

As a result, I try to set grid size as 3,4,5,6,7 in the following study to find the

optimal parameter value. I still use the grid search method, which gives me a good hint of the value. To use a most moderate value, my best guess for the optimal paramater is around grid = 5 and Learning rate = 0.15. Also, I would like to use a dynamic learning rate to improve the model. Setting a large learning rate will help the model to converge faster at the beginning and reducing the learning rate will helps the model to be more stable in the end. So in the end, my choice is to set initial learning rate as 0.15 and gradually reduce it to 0.01.

However, even though we dramatically increases the evaluation value, since grid size is too small, it falls into the same issue as DBSCAN model and spectral clustering model that it ignores some minor differences and treats majority of the data in one cluster as shown in the following graph.

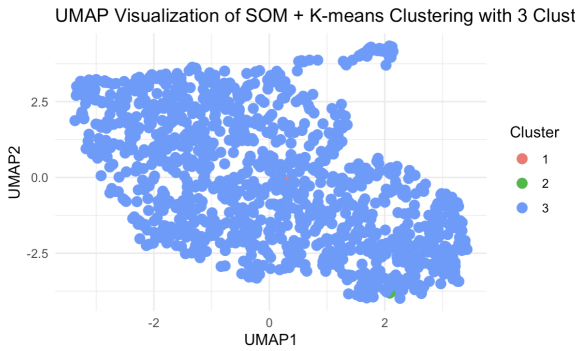


Figure 4.2: SOM with 5 GD and initial 0.15 LR

As a result, I go back to the origin and do a more comprehensive grid search for the parameter. To avoid the previous problem, I set the grid size from 10 to 20. I generated a heatmap to visualize the grid search result table.

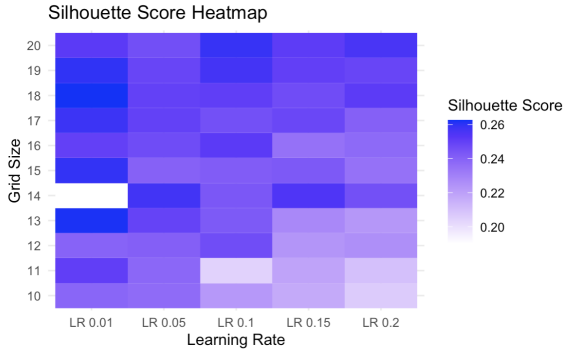


Figure 4.3: Enter Caption

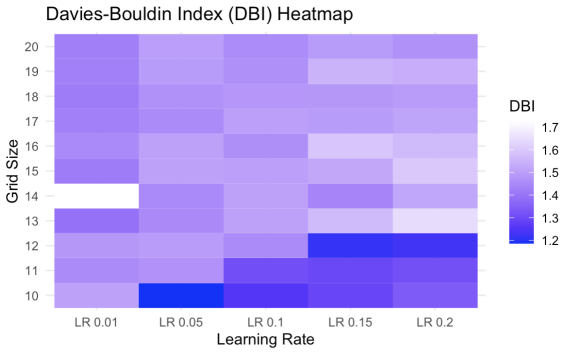


Figure 4.4: Enter Caption

From these two heatmaps, we can find that silhouette score and DBI shows different preference to the parameter setting. Silhouette score prefers a higher grid size with minimum leanrning rate, while DBI prefers a lower grid size with maximum learning rate. Here we have two considerations: First, Silhouette score shows a absolutely strong pattern, but DBI is still acceptable at the left top of the heatmap. Second, silhouette score is a more important evaluation score than DBI. As a result, I would set our following models at grid size = 18 and learning rate = 0.01. Our original model used a default setting to set $GridSize =^{2.5} \sqrt{N} = 1.7386$. As a comparison, the original performance table is like figure 24. By setting Grid Size = 18 and LR = 0.01, the model improved its silhouette score from 0.2565725 to 0.2626407, around 4 percent increase.

SOM Evaluation Metrics

k	silhouette_score	dbi_value
2	0.2452293	1.622606
3	0.2565725	1.439141
4	0.1985729	1.531510
5	0.1948893	1.505420

Figure 4.5: Default SOM performance table

SOM Evaluation Metrics

k	silhouette_score	dbi_value
2	0.2514851	1.561559
3	0.2626407	1.405807
4	0.2079783	1.478796
5	0.1948963	1.502243

Figure 4.6: GS 18 LR 0.01 SOM performance table

4.2 Reducing features

Even though ChatterBaby group has already done feature selections, using PCA might still able to improve cluster separation since it can future reduce the existence of noise.

As a result, continuing to use Grid size = 18 and LR = 0.01. Here is the plot showing how PCA can affect our evaluation matrices.

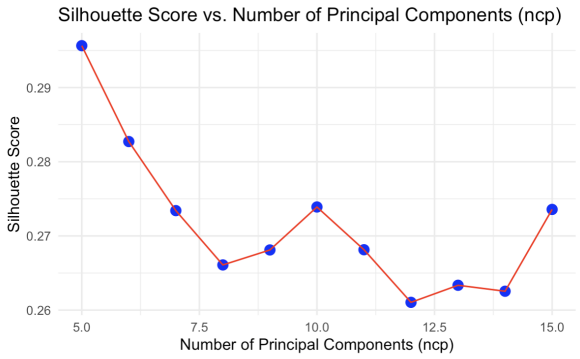


Figure 4.7: PCA ncp value vs silhouette score

Knowing the fact that limiting ncp value too much will result in a decrease in the information kept in the model, which is a situation that we are not willing to see, I would like to choose ncp = 10 as our target since it is a local peak value that is not too extreme.

In the this time, we increase our model's silhouette score from 0.2626407 to 0.275633, around 5 percent increase.

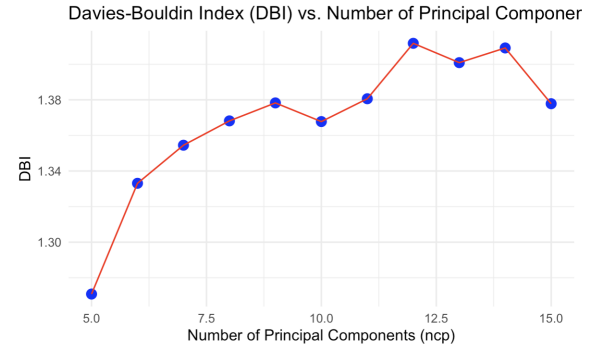


Figure 4.8: PCA ncp value vs DBI

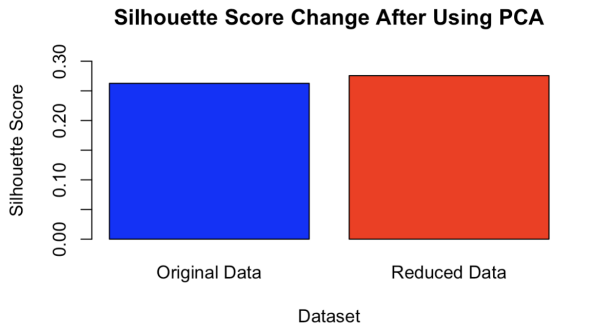


Figure 4.9: PCA after and before (Silhouette score)

4.3 Final SOM model with k-means k = 3

To begin with, here is the plot of our final model UMAP visualization.

Like we've discussed in previous section, the model separates two major clusters along with one minor but not sparse cluster. Our model reaches a Silhouette score of 0.275633 with a DBI of 1.367770.

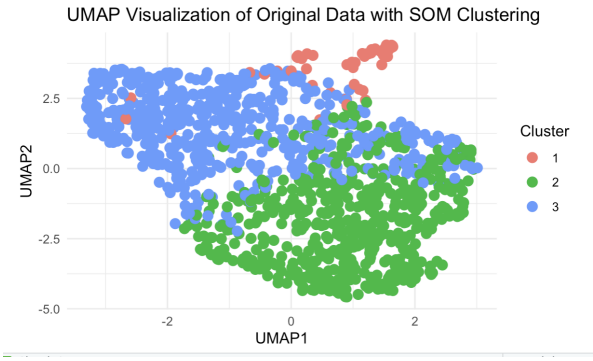


Figure 4.10: Final SOM UMAP visualization

CHAPTER 5

Validating Model With Test Dataset

In this chapter, I will use the test dataset to test our model to see if it can function generally as we expected from the training section.

5.1 Introduction of the test dataset

Like the training dataset, testing dataset is also pre-cleaned and organized by ChatterBaby. It contains 1250 observations, the same amount as my training dataset. The dataset remains blocked until I finish modifying my model, which means that nothing is changed after we use test dataset to test the performance of the model.

5.2 Final model evaluation

In this section, we will make a comprehensive analysis of my model. I'll start from comparing the clustering result with training dataset to see if they are homogeneous, and then talk about the evaluation matrices of the test dataset.

5.2.1 Comparison between test and training result

From the two plots, we can clearly see that two matrices are showing similar distribution patterns. Especially we can see that original pain cluster and fussy cluster shows opposite preference in fitting themselves into cluster 2 and 3, which is a good thing for us to move further.

However, as I compute the chi-square test between test and train dataset to verify if they are homogeneous, the test result shows statistically significant difference between

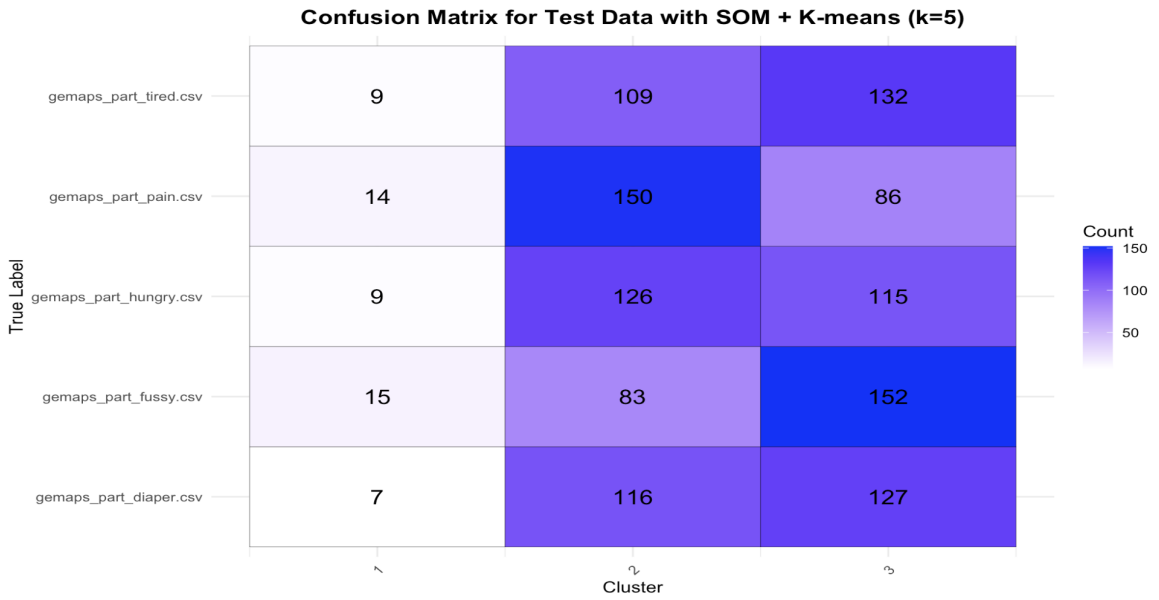


Figure 5.1: test confusion matrix

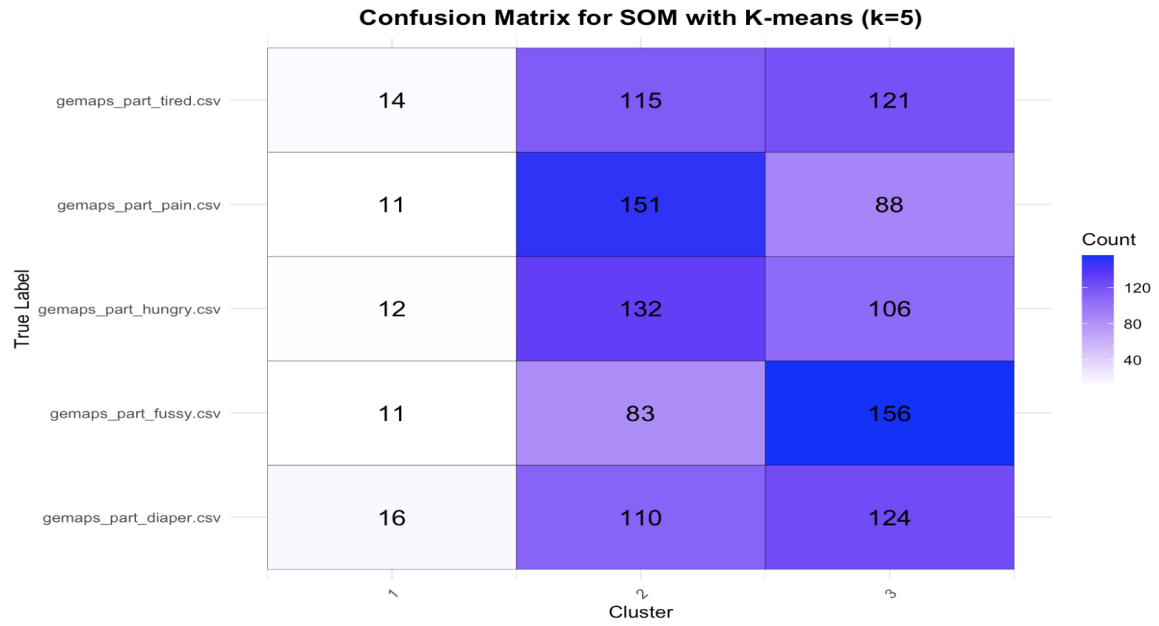


Figure 5.2: train confusion matrix

these two groups. It shows that we should not generalize our model to a broader dataset. The difference between the minor visual differences between two confusion matrices and the low P-value from chi-square test is a topic that worth futher study.

```
        Pearson's Chi-squared test

data:  combined_cm
X-squared = 88.907, df = 20, p-value = 1.15e-10
```

Figure 5.3: Enter Caption

5.2.2 Evaluation

Then how exactly our test result performs? Unfortunatly, the test result is not as promising as I expected. The result is just better than original K-means and hierachical clustering result but it didn't provide us enough confidence to conclude the effectiveness of the model.

```
Test Silhouette Score: 0.1830287
Test Davies-Bouldin Index (DBI): 2.272433
Test Purity: 0.2536
```

Figure 5.4: Test evaluation result

5.2.3 UMAP Visualization

Here is the UMAP visualization of the test result. Still, there are two main clusters with one minor cluster.

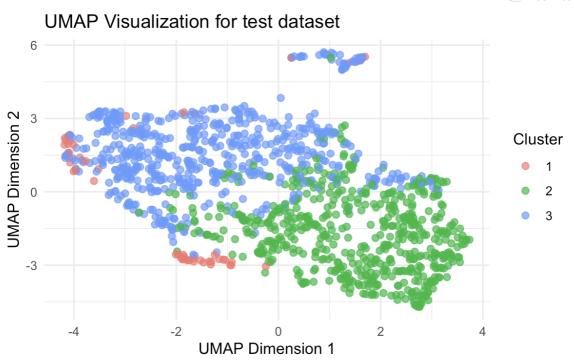


Figure 5.5: Umap Visualization for test result

CHAPTER 6

Conclusion

In this chapter, I'll provide a conclusion for the whole thesis and provide some future work direction for the following researchers.

6.1 Conclusion

In this thesis, we applied multiple unsupervised machine learning techniques, including K-means, DBscan, hierarchical clustering, self-organizing map and spectral clustering, to detect the optimal clustering method for a baby cry dataset of acoustic features. Among all the models, our study shows two kinds of clustering results.

The first type of result shows an extremely good evaluation performance with a tendency to put almost all data points into one cluster while leaving the rest clusters addressing outliers. Since our goal is to provide some useful clustering insight for a group developing models detecting the reason of baby crying, it make minimum contribution to the group by telling them most of the babies are crying for the same reason. As a result, I would drop those results even though their silhouette score can be very high.

The second type of result presents a relatively weak evaluation performance with a more evenly distributed clustering result. This type of result is gained by K-means clustering, Hierarchical clustering and SOM followed by K-means clustering method. After comparison, we conclude that SOM followed by K-means clustering has the best performance.

In order to fully explore the potential of SOM model, I tried to fine-tune the parameters of SOM model and also tried to use PCA to reduce the complexity of the dataset. In the end, we set $gridsize = 18$, $learningrate = 0.01$ and $ncp = 10$ for our model,

which did a not satisfying job for the test dataset. But the confusion matrix still gives me confidence to say that decreasing the number of clusters are reasonable and worth keep studying.

Returning to the origin of our study, I would recommend Dr. Anderson to reduce the number of clusters from 5 to 3 to reduce the overlapping problem among baby cry data.

6.2 Future Work Direction

Here are few things I notice in the study.

First, like I've indicated in previous section, there is one type of clustering result suggesting all the data points should be in one cluster. Some scholars hold similar view points that babies might simply cry because they are feeling uncomfort, however babies don't have the ability to distinguish if that uncomfort is caused by hungry, pain, or something else. So it is not guaranteed that we can gain enough valuable information from the acoustic features to detect the reason behind baby crying.

Second, as I revisit the models I use, I realized that I was using SOM pre-operated data to calculate silhouette score and DBI in the training part, which likely results in the sudden decrease of the performance in test dataset since it requires to use original data to calculate indeces instead of SOM pre-operated ones. Future researchers are encouraged to use original data points throughout the training to make the study more accurate.

Third, there is still lots of potential to discover for SOM model. For example, we only used K-means to finalize the model. Future researchers can try applying more advanced clustering methods to do the second part of SOM model and see if it can improve the results.

REFERENCES

[Bum] The Bump. “11 Reasons Why Babies Cry—And How to Soothe Their Tears.”.

[Cha] ChatterBaby. “ChatterBaby website.”.