Data mining: fundamentals

EXAMINATION PROJECT REPORT

Can raising voice really help in expressing emotions?

Assessing relationships between vocal strength and emotionality

Lucrezia Labardi (600163)[1]
Vincenzo Sammartino (599203)[1]
Daniele Borghesi (578406)[2]

[1]*Master's degree in Digital Humanities, Università di Pisa*
[2]*Master's degree in Data Science and Business Informatics, Università di Pisa*

ACADEMIC YEAR 2022-2023

Contents

1 Introduction

Every time someone talks, either through a speech or a song, a great variety of information is delivered, and every piece of it can be used by the listener to detect what it was trying to communicate. This project focuses on exploring the relationship between the intensity of the voice, understood as the volume of sound, and the emphasis of an emotion.

The exploited dataset is composed of *RAVDESS*'s audio records, containing various information related to the soundtrack of some utterances: each utterance was pronounced by simulating a specific emotion. The data were explored, prepared, and cleaned (section 2), and then analyzed and divided into clusters by exploiting numerous clustering algorithms (section 3). Several classification operations (section 4) were also performed, exploiting Machine Learning algorithms. Finally, an exploration of the most frequent patterns in the dataset was performed, identifying rules associated with the recognition of each specific emotion (section 6).

Conscious of the fact that the expression of emotion varies according to the mood and attitude of the individual, this project seeks to understand whether raising or lowering the voice (and thus the intensity) can help to better recognize certain emotions. For this reason, emotions were divided into *strong emotions* and *weak emotions* to ensure a better analysis (see section 2.1).

In the end, since *RAVDESS* dataset is often used to determine a specific emotion, the intention was to add a slightly new point of view to the existing related works.

2 Data Understanding and Preparation

As anticipated, the dataset that will be analyzed in the present paper is the *Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)*. The original dataset files are divided into three modalities, but we are going to consider only the Audio-Only ones. The database consists of 24 professional actors (12 male and 12 female), each performing vocalizations of two statements with emotions that include: *happy*, *sad*, *angry*, *fearful*, *surprise*, *disgust*, *calm*, and *neutral*.

The categorical feature that occurred to be important for our analysis goal is `emotion` for the emotion conveyed. Alongside this, the database also consisted of many continuous features for describing the audio files: `length_ms` for the length expressed in milliseconds and `intensity` expressed in dBFS. Principal statistics (`min`, `max`, `mean`, `std`, `kur`, `skew`) were given both for the entire dataset and for *Mel-Frequency Cepstral Coefficients* (MFCC),[1] *Spectral Centroid*,[2] and *Short-Time Fourier Transform Cromagram* (STFT).[3]

Data Understanding and Preparation is one of the most important parts of a Data Mining project, and for this reason, it was given a lot of space in this work. This section shows how raw data were transformed handling some specific problems of data cleaning: the presence of missing values and irrelevant features (section 2.2), and outliers (section 2.5). The dataset didn't have either duplicate data or syntactic errors. Some categorical analyses are here reported to explain how outliers were identified and why it was decided to handle them in a certain way (section 2.3). The actual analysis began right after the preparation step and initially, it consisted in observing correlations that emerged between the selected data (section 2.4).

2.1 Subdivision of data

As already explained, one of the goals of this project is to test whether there is a relationship between intensity (understood as voice volume) and emphasis of an emotion. In other words, to test whether using a higher or lower tone of voice can promote the recognition of certain emotions. Regardless of data analysis, it is possible to intuitively divide emotions into two broad categories:

- *strong* emotions: *anger*, *surprise*, *fright*, *happiness* and *disgust*
- *weak* emotions: *neutrality*, *sadness* and *calm*

[1] Mel-Frequency Cepstral Coefficients is a mathematical method which transforms the power spectrum of an audio signal to a small number of coefficients representing the power of the audio signal in a frequency region (a region of pitch) taken for time. MFC coefficients give us an idea of the changing pitch of an audio signal.

[2] Spectral Centroid is a measure used to characterize a spectrum. It indicates where the center of mass of the spectrum is located and it is connected with the impression of the brightness of a sound.

[3] Short-Time Fourier Transform is a measure that cuts audio waveform into short, overlapping equal-length segments and takes the Fourier transform of each segment individually to produce multiple power spectrograms, identifying resonant frequencies. Its major advantage is a better resolution of changes in the audio signal compared to time.

For emotions commonly regarded as 'strong', it is possible to intuitively associate a high emotional emphasis with a higher sound intensity (a higher voice volume); conversely, for emotions commonly regarded as 'weak', a high emotional emphasis could be associated with a lower sound intensity (a lower voice volume).

On the basis of these assumptions and observations, the main dataset (2452 elements) was used to generate two new datasets:

- an *high emotional emphasis* dataset (1318 elements), containing records of "strong" emotions with an intensity above the median of that specific emotion, and record of "weak" emotions with an intensity below the median of that specific emotion.
- a *low emotional emphasis* dataset (1134 elements), with all remaining data

While the data understanding and preparation operations were performed on the complete dataset, the subsequent clustering and classification operations were done by comparing the behavior of the algorithms on all 3 datasets.

2.2 Filling in missing values

The dataset presented missing values, denoted as *NaN*, only in three columns: 1130 in *actor*, 196 in *vocal_channel*, and 816 in *intensity*. The values in the first two columns are categorical while the values in the third column are numerical. For this reason, the problem was handled following the same logic but in a slightly different way. To perform the substitution in columns *actor* and *vocal_channel*, it was applied an algorithm which followed these main two steps:

1. The dataset records were grouped by *emotional_intensity*, *sex*, *statement*, and *repetition*. *Emotion* was excluded, as it is involved in the subsequent classification and regression tasks;
2. In each grouping, each *NaN* value was replaced by randomly drawing a value from among the non-*NaN* values. The casual extraction is needed to maintain the same odds ratio between the values in the column.

In the case of continuous values, like *intensity*, the algorithm is similar, but the substitution was calculated using the median of the values of the relative groupings.[4] At the end of this phase, it was obtained a dataset in which all columns are complete.

2.3 Categorical data analysis

During the categorical analysis, the characteristics of values of each emotion for the eight main continuous features - *intensity*, *zero_crossings_sum*, *max*, *min*, *skew*, *std*, *kur*, *length_ms* - was displayed and compared according to *vocal_channel* [speech, song], *emotional_intensity* [normal, strong] and *sex* [M, F] values.

Several pieces of evidence emerged from the graphs, here the main ones are briefly outlined:

- In every category the emotion *angry* has the highest intensity, while *calm* has the lowest. Typically songs have a higher intensity than speeches, even *calm*, *neutral*, and *sad*. Wherever *emotional_intensity* is strong, emotions which have higher intensity are *fearful*, *happy*, and *angry*. This behavior is less visible in emotions like *surprised* and *disgust*, and it isn't visible in the others.
- The values of the `zero_crossings_sums`[5] are lower in the male for every emotion. In women, the distribution is more normal.
- *Fearful*, *angry*, and *happy* emotions have a wider range of values and a more normal distribution for `max` feature. On the contrary, *normal* has a narrower range of values and has more spikes in the distribution. For the `min` feature the behavior is the same but specular to the x-axis. This is strongly connected with the standard deviation's values, which for low-emphasis emotions is close to 0.
- Standard deviation (`std`) is typically lower in speeches than in songs. In *calm*, *neutral*, and *sad* there are more spikes while other emotions have a more normal distribution.

[4] The filling of missing values in *vocal_channel* and *actor* was performed before *intensity*, to enable us to use *vocal_channel* as a grouping attribute for *intensity* in step 1, increasing the precision

[5] The zero-crossing sum is the summation of the zero-crossing rates, the rate at which a signal changes from positive to zero to negative or from negative to zero to positive.

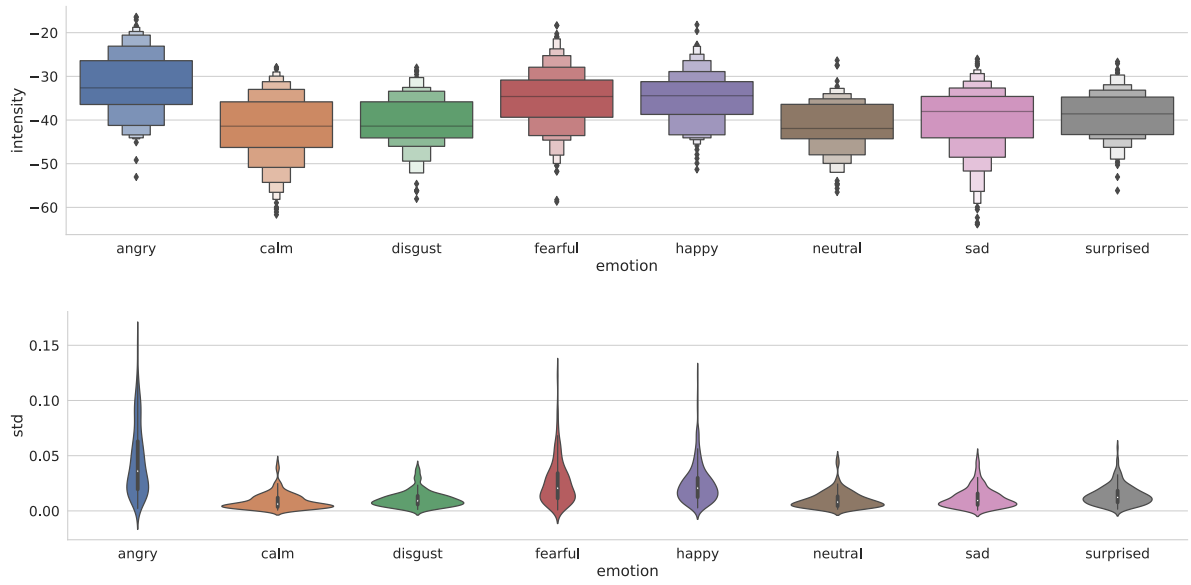

Figure 1: Comparison between the *Intensity* (up) and the Standard Deviation (*Std*) of each emotion

- For Kurtosis (`kur`)[6] values, songs have lower values than speeches in every emotion.

2.4 Correlations and irrelevant attributes

Irrelevant attributes are the ones that contain almost no useful information for the data mining tasks. In this case, some attributes were found to have the same values for each row: *modality* with "audio_only" value, *frame_rate* with "48000", *sample_width* with "2" and *stft_max* with "1.". Since redundant and irrelevant attributes can reduce classification accuracy and the quality of the clusters, these attributes were deleted from the dataset. As a result of this operation, the number of attributes in the dataset was reduced from 38 to 34.

Below, we analyzed possible correlations: statistical analysis used to measure the relationship between two features. Correlations in our dataset were explored using the standard *Pearson's correlation coefficient*. Figure 2 displays a heatmap where color intensity indicates a greater correlation between two attributes. Since the correlations are symmetrical to the diagonal of the matrix, the image shows only the half below the diagonal.

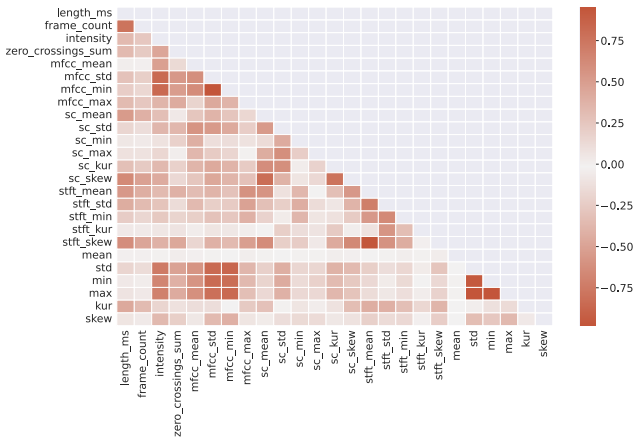

Figure 2: Heatmap of correlations between attributes

It is evident that there are many attributes with a very high positive or negative correlation and

[6]Kurtosis is a measure of the "tailedness" of the probability distribution of a real-valued random variable.

therefore, offering correlated and very similar information. Between two highly related attributes that provide the same information, one can be removed to reduce the size of the dataset. For each correlation greater than 0.75 or less than -0.75, it was decided to eliminate the corresponding attribute on the y-axis (the rows of the matrix shown in figure 2). Through this operation, it was possible to remove 9 attributes: *frame_width*, *frame_count*, *mfcc_std*, *mfcc_min*, *sc_skew*, *stft_skew*, *std*, *min*, *max*. The number of attributes in the dataset was reduced from 34 to 25.

2.5 Handling of outliers and skewness

The task of identifying outliers (also called "anomaly detection") consists in identifying observations whose characteristics are significantly different from the rest of the data. The goal is to discover real outliers and avoid falsely labeling normal objects as anomalous. The removal of outliers will also improve the distributions of some attributes, decreasing their skewness. These steps will be useful for subsequent clustering and classification.

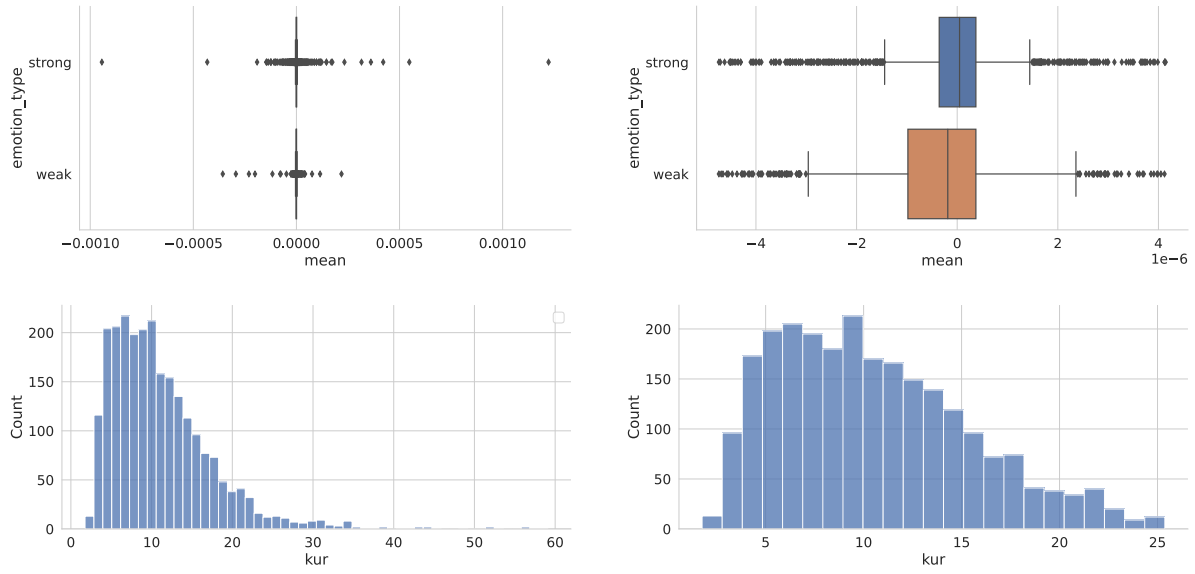

Figure 3: Comparison between distributions of *Mean* (up) and *Kur* (down) attributes with (on the left) and without outliers (on the right)

The presence of outliers in the dataset was well visible for several features: an example of the feature `mean` is shown in figure 1. Outliers are often handled with deletion but this was not the case: was implemented an algorithm which found the values which lay in the first and the fourth quartile - i.e. the values that lie below 25% and above 75% of the distribution - and replace them with the median of a specific grouping they belonged to, following the same logic as for the replacement of missing values (see section 2.2). Some examples of the result of this operation are shown in figure 1 and 3: it is well visible how data became more normally distributed, free of anomalous elements.

3 Clustering

This section illustrates the behavior of different clustering algorithms on the available data. Specifically, three different clustering families are compared: *Centroid-based* clustering, *Density-based* clustering, and *Agglomerative* clustering.

The aim is to understand whether the data have natural clusters and whether these clusters can be matched, even partially, to the data of each emotion. The algorithms were compared with the three different types of datasets illustrated in section 2.1 (*complete*, *high emphasis* and *low emphasis*), to understand, even before classification operations, whether emotions can be accurately distinguished even without high emphasis.

3.1 Attribute selection and standardization

An excessive number of attributes may be an obstacle to the efficiency of clustering algorithms. These tend to benefit from a dimensionality reduction, and from a selection of the most relevant attributes. In this regard, it was decided to exploit Principal Component Analysis (PCA) to establish which features determine the greatest weight in each Principal Component. Specifically, it was decided to reduce the dimensionality to 3 features.

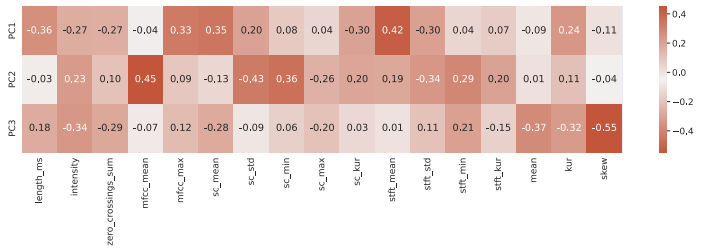

Figure 4: Weight of the attibutes in each Principal Component

As it's shown in figure 4, each attribute has an importance score in each Principal Component: *stft_mean* is the most important feature in PC1, *mfcc_mean* is the most important feature in PC2, and *skew* is the most important feature in PC3. Then it was decided to reduce all the datasets to these three features for testing all clustering algorithms.

Finally, all data were standardized by removing the mean and scaling to unit variance to further facilitate clustering and classification operations.[7]

3.2 Centroid-based methods

This section shows the various centroid-based clustering methods: KMeans (section 3.2.1), Bisecting KMeans (section 3.2.2) and X-Means (section 3.2.3).

3.2.1 K-Means

Before analyzing the results from the execution of K-Means on the various datasets, it is necessary to carry out some evaluation and optimization of the algorithm, to determine the most suitable number of centroids to obtain the best possible cluster subdivision.

In this regard, it was decided to perform the optimization of the number of centroids only on the complete dataset, containing both high- and low-emphasis records. Two specific metrics were used: the *Silhouette score* and the *Sum of Squared Errors* (*SSE*). The metrics were measured for the execution of the K-Means with a minimum of 2 and a maximum of 16 clusters (the number of emotions, 8, multiplied by the 2 levels of emotional emphasis).

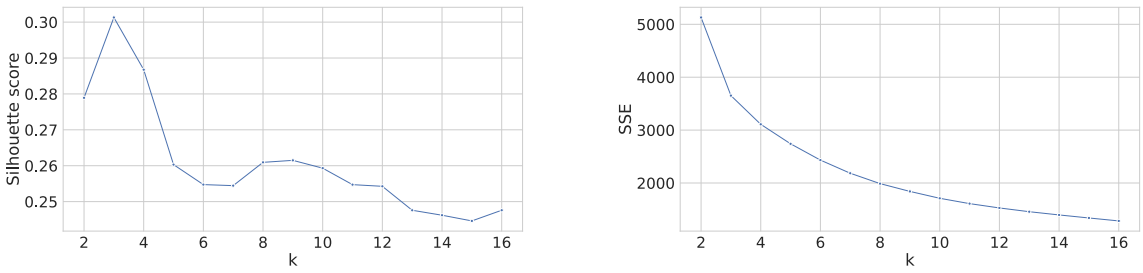

Figure 5: Trend of Silhouette score and ESS as k value (number of centroids) increases in K-Means

As can be seen from figure 5, the two indicators used suggest two different options for choosing the number of centroids: in the case of the Silhouette score, the highest value (0.30) is reached at 3 centroids; in the case of SSE, this continues to decrease as the number of centroids increases, in a progressively slower way. As noted, in this case, the SSE does not provide us with an accurate indication, as the illustrated

[7] `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html`

Table 2: Silhouette Score and SSE for some k values (number of centroids) with K-Means (2-6)

Score	**2**	**3**	**4**	**5**	**6**
Silhouette	0.28	0.30	0.29	0.26	0.25
SSE	5129.49	3652.64	3109.66	2741.12	2434.92

curve does not show a sufficiently marked flattening to determine the optimal number of centroids. In contrast, the Silhouette score shows a clear peak in the proximity of 2, 3, and 4 centroids. It is then decided to proceed in the experiments using 4 centroids, following a trade-off between both indicators.

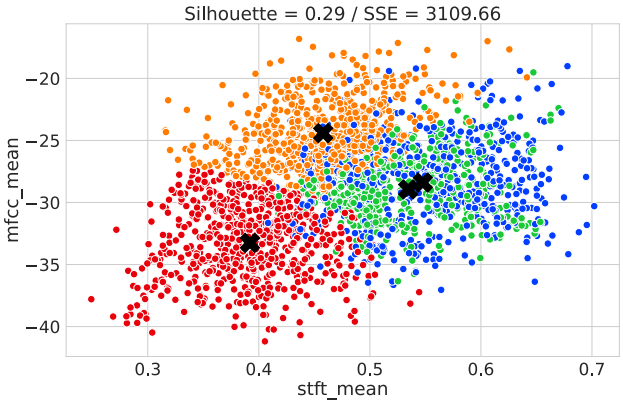

Table 3: Dimension of the clusters with K-Means

Dataset (emphasis)	1	2	3	4
Both	616	685	454	697
High	304	457	329	228
Low	254	321	297	262

Figure 6: Clustering obtained with K-Means on the complete dataset (4 centroids)

Figure 6 shows the two most relevant attributes of the dataset (*stft_mean* and *mfcc_mean*), according to PCA. The K-Means algorithm, configured with 4 centroids, showed 4 distinct clusters: restricting only to two dimensions, it is possible to determine how the clusters delimited by the algorithm appear to be distinct and not very overlapping.

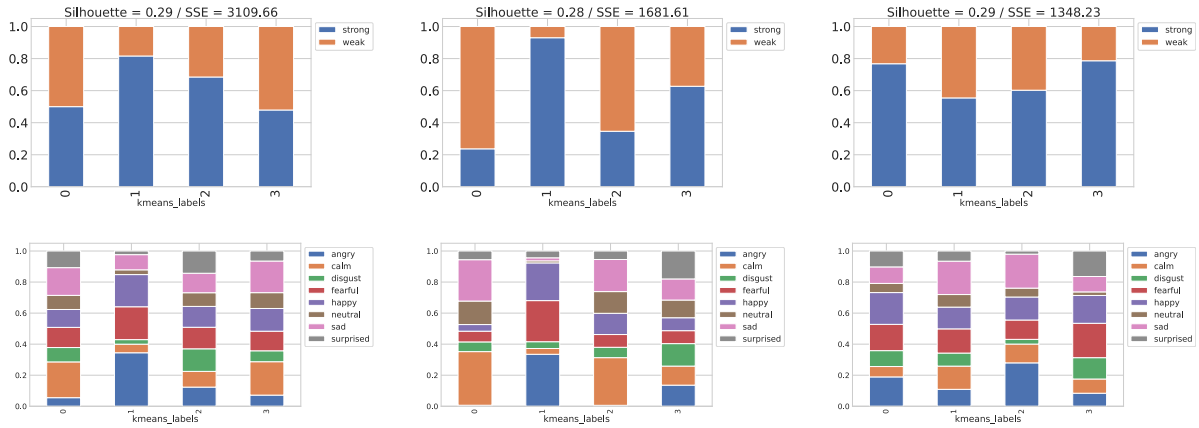

Figure 7: Emotion type (up) and emotions (down) clustering characterization with K-Means

Figure 7 shows two groups of bar charts, respectively for emotion type and single emotions. These charts illustrate how well the clusters are able to represent individual emotions and the corresponding categories. It is evident that in the dataset containing data for both levels of emphasis, K-Means was not particularly effective in clustering emotions: with the exception of the second and third clusters (clusters 1 and 2), where a greater presence of strong emotions is evident, the remaining two clusters agglomerate emotions of both categories in apparently equal amounts. This phenomenon is also evident when looking at individual emotions, rather than emphasis categories: the first and the fourth clusters (clusters 0 and 3) do not have a preponderance of 'strong' or 'weak' emotional type, showing only small differences between the individual emotions. Very similar behavior occurs when using low emphasis.

The behavior is clearly different in the high-emphasis dataset: in this case, the algorithm is able to

cluster emotional types more accurately than in the other two scenarios. Indeed, the first and third clusters (clusters 0 and 2) show a clear majority of data from weak emotion type, whereas the second cluster (cluster 1) is almost entirely populated by strong emotion type data. Looking also at the graph of individual emotions, it is clearly evident that emotions such as anger, fear, and happiness are in a clear majority in comparison to the others in the second cluster, while emotions such as calm, sadness, and neutrality are preponderant in the first and the third cluster.

These first results may already show us how a higher emotional emphasis can actually make a clustering process more successful in recognizing emotions. It is also interesting to observe how the emotions of surprise and disgust prove to be more complex to distinguish for K-Means, even in the case of high emotional intensity: these emotions, in fact, show an almost similar presence in all clusters, posing themselves as ambiguous.

3.2.2 Bisecting K-Means

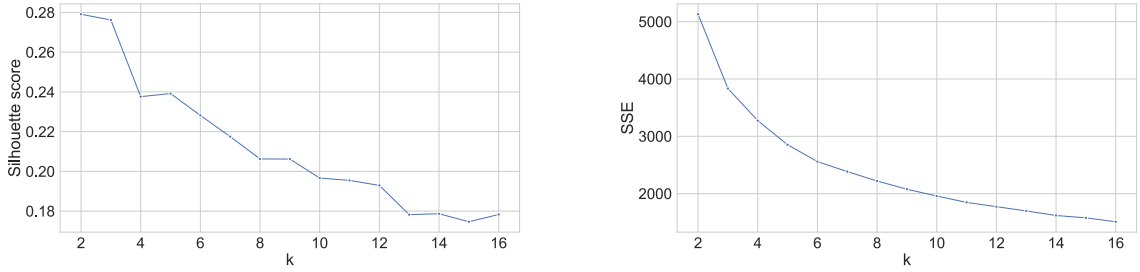

Figure 8: Trend of Silhouette score and ESS as k value (number of centroids) increases with Bisecting K-Means

Table 5: Silhouette Score and SSE for some k values (number of centroids) with Bisecting K-Means (2-6)

Score	**2**	**3**	**4**	**5**	**6**
Silhouette	0.28	0.28	0.24	0.24	0.23
SSE	5129.49	3834.58	3273.18	2853.17	2558.98

With Bisecting K-Means, looking at figure 8 it can be seen that an increase in the number of centroids corresponds to a decrease in the SSE value, but also to a decrease in the Silhouette score. On the basis of the results, it was decided to use a number of centroids equal to 3, in order to keep the Silhouette value as high as possible, but without sacrificing too much SSE. It can already be seen that the Silhouette score is slightly lower than with K-Means.

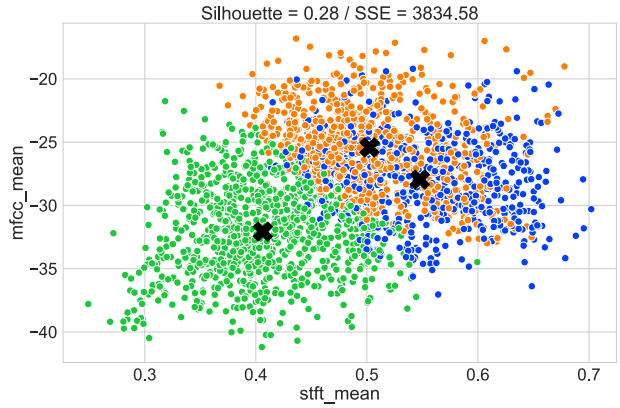

Figure 9: Clustering obtained with Bisecting K-Means on the complete dataset (3 centroids)

Table 6: Dimension of the clusters with Bisecting K-Means

Dataset (emphasis)	1	2	3	4
Both	616	685	454	697
High	304	457	329	228
Low	254	321	297	262

As already observed for the K-Means algorithm, figure 9 shows three distinguishable clusters of similar size (see table 6). Again, however, the characterization of the clusters needs to be analyzed in more depth to assess their quality and purity.

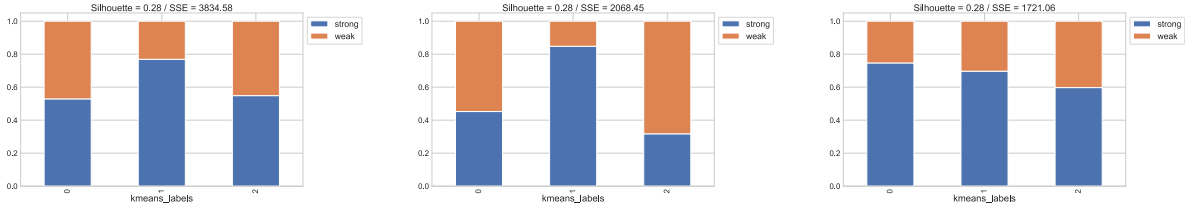

Figure 10: Emotion type clustering characterization with Bisecting K-Means

The behavior that emerges when observing the bar charts illustrated in figure 10 is also extremely similar to that seen with K-Means: the algorithm performed on the dataset with high emphasis records obtains purer clusters than those obtained in the other two datasets. In the high-emphasis dataset, the first and the third clusters tend to be largely populated by data relating to weak emotions, while the second cluster is almost entirely composed of data relating to strong emotions.

3.2.3 X-Means

In the case of the X-Means algorithm, it is necessary to specify that the maximum number of clusters that can be allocated (`kmax`) was set to 16, maintaining the same logic used for K-Means. As a criterion for splitting (`criterion`), the Bayesian Information Criterion (BIC) was used.

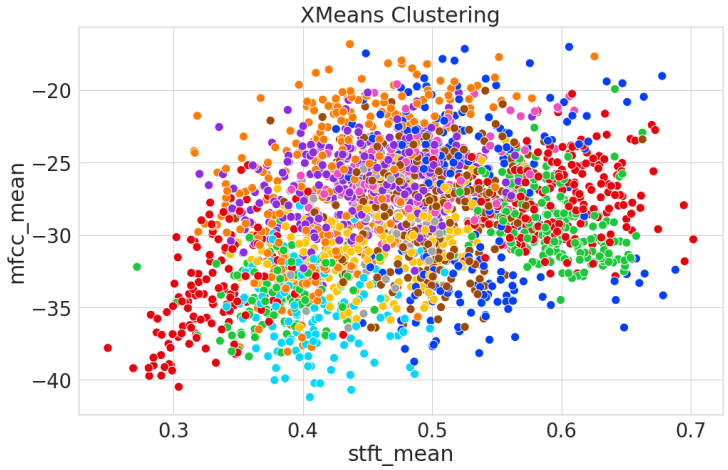

Figure 11: Clustering obtained with X-Means on the complete dataset (16 centroids)

For this clustering algorithm, performed on the complete dataset, the number of clusters obtained by testing the entire dataset is 16, which is the maximum value set. Image 11 shows that the clustering seems confusing, and this is also reflected in the data, showing a low Silouhette value and an excessively high SSE value in comparison to K-Means and Bisecting K-Means.

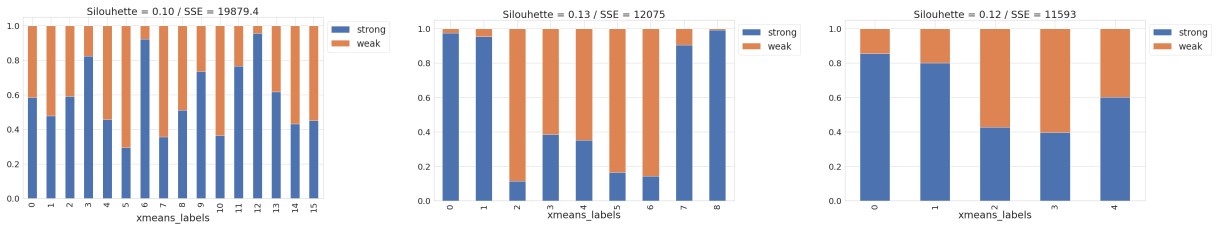

Figure 12: Emotion type clustering characterization with X-Means

The charts in figure 12 show that clusters are not well divided in the case of the complete dataset. The

situation changes radically in the case of the high-emphasis dataset: a decisive improvement in emotion type clustering can be seen. Despite lower Silhouette values and higher SSE values, the graph seems to show more accurate clustering than observed with K-Means and Bisecting K-Means, even in the case of the low-emphasis dataset.

3.3 Density-based clustering

In the second instance, it was decided to test the behavior of density-based clustering algorithms: specifically, the DBSCAN algorithm and the OPTICS algorithm were chosen. Again, the behavior of the algorithms was verified on all three generated datasets (*complete*, *high-emphasis* and *low-emphasis*), for a more accurate comparison.

3.3.1 DBSCAN

As already observed with the centroid-based algorithms, also the DBSCAN algorithm has parameters that can be optimized to find the best clustering configuration. To be precise, it was chosen to perform a cross-comparison between a set of *epsilon* values (the maximum distance between two samples for one to be considered as in the neighborhood of the other) and a series of values to set the number of *minimum samples*: the number of samples in a neighborhood for a point to be considered as a core point.

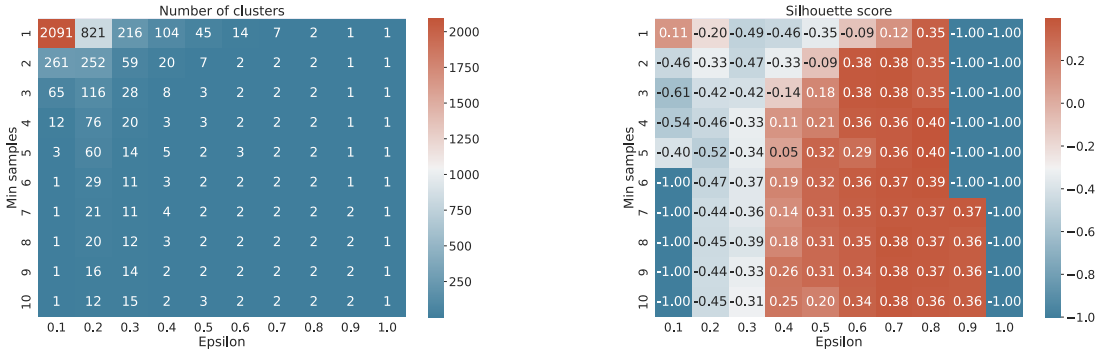

Figure 13: Tuning of the epsilon parameter and the number minimum samples with DBSCAN

As illustrated in figure 13, it is evident how, in the case of DBSCAN, a greater number of clusters leads to a drastic lowering of the Silhouette score, much more severe than seen with K-Means. Excluding the cases where only one cluster could be generated, the highest average Silhouette scores are achieved in correspondence of 2 clusters: to be precise, using an *epsilon* value of 0.8 and a *minimum samples* of 4 or 5. We choose to proceed using this configuration: *epsilon* 0.8 and 4 *minimum samples*.

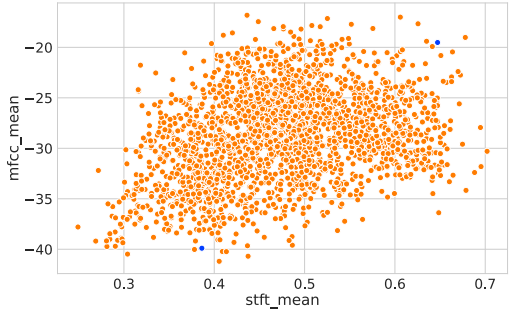

Figure 14: Clustering obtained with DBSCAN on the complete dataset (4 minimum samples and epsilon 0.8)

Table 7: Dimension of the clusters with DBSCAN

Dataset (emphasis)	**1**	**2**
Both	2	2450
High	2	1316
Low	7	1127

As can be seen from figure 14, in this case, the Silhouette score (although higher than that obtained with K-Means) did not result in an optimal cluster configuration: as also indicated in table 7, the results show a single large cluster comprising almost all the points, and the second cluster of insignificant size (2

points). Even without deepening the characterization of emotions (as seen in the centroid-based clustering algorithms), it is evident that the 2 clusters generated cannot correspond to either the emotional groups or their types (strong or weak).

Even if not illustrated in the present figures, also by choosing a configuration with a greater number of clusters, the clusters obtained fail to characterize the emotions or the types to which they belong, without considering a drastic drop in the Silhouette score.

3.3.2 OPTICS

As already performed for DBSCAN, also for the OPTICS algorithm it was decided to optimize the parameters already seen previously (*epsilon* and *minimum Samples*), through a cross-comparison with the number of clusters generated and the corresponding Silhouette Score.

The optimization results are very similar to those visited for DBSCAN (see figure 13): also, in this case, an increase in the number of clusters corresponds to a drastic drop in the Silhouette score. Based on this one, the best score (0.40) is obtained using an *epsilon* equal to 0.8, and a number of *minimum samples* equal to 4: exactly the same configuration used for DBSCAN. The clusters obtained through this configuration of the algorithm turned out to be exactly identical to those obtained through DBSCAN (see figure 14 and table 7), determining how also the OPTICS algorithm is not able to correctly characterize the emotions or their belonging categories.

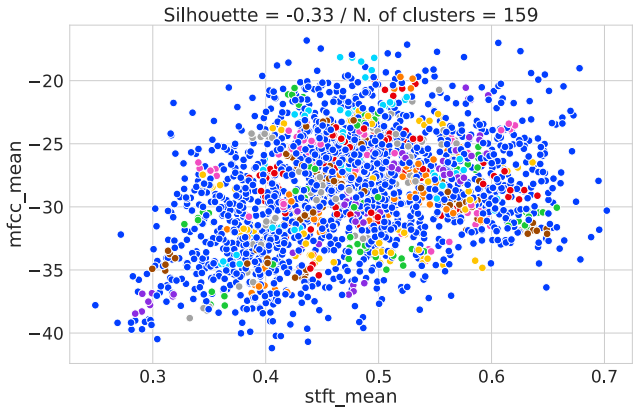

Figure 15: Clustering obtained with OPTICS on the complete dataset (without maximum epsilon)

A second experiment was performed without setting an *epsilon* value, and thus setting the maximum *epsilon* (maximum distance between two samples for one to be considered as in the neighborhood of the other) to infinite, allowing the algorithm to identify clusters across all scales. As can be seen from figure 15, the algorithm again obtained clusters unsuitable for our task. The high number of clusters (159) led to a drastic drop in Silhouette's score (-0.33), without approaching a cluster configuration similar to that seen with K-Means, for example.

3.4 Agglomerative clustering

Agglomerative clustering (or hierarchical clustering) can be used with four different techniques of agglomeration of clusters (types of distance to use between sets of observations): *single*, *complete*, *group average* and *Ward*. It was therefore decided to test, using the dataset containing both levels of emphasis, all four configurations, to identify the most promising one.

As illustrated in figure 16, a dendrogram was extracted for each type of linkage. The dendrograms obtained differ considerably from one another: the first, corresponding to the single link configuration, does not appear to show the presence of large, distinct, and separate clusters in the data.

The situation differs considerably in the case of the other types of linkage: in all three cases, it is possible to note how the algorithm manages to generate larger clusters already at relatively closer distances. In particular, the algorithm performed with Ward's technique reveals the formation of 5 clusters of relevant size just above the distance threshold of 20 (out of a maximum distance of more than 60) and the agglomeration of 3 clusters close to the distance of 30. Subsequently, the algorithm does not start to link these 3 clusters until it reaches the distance threshold of 50, almost at the limit of the

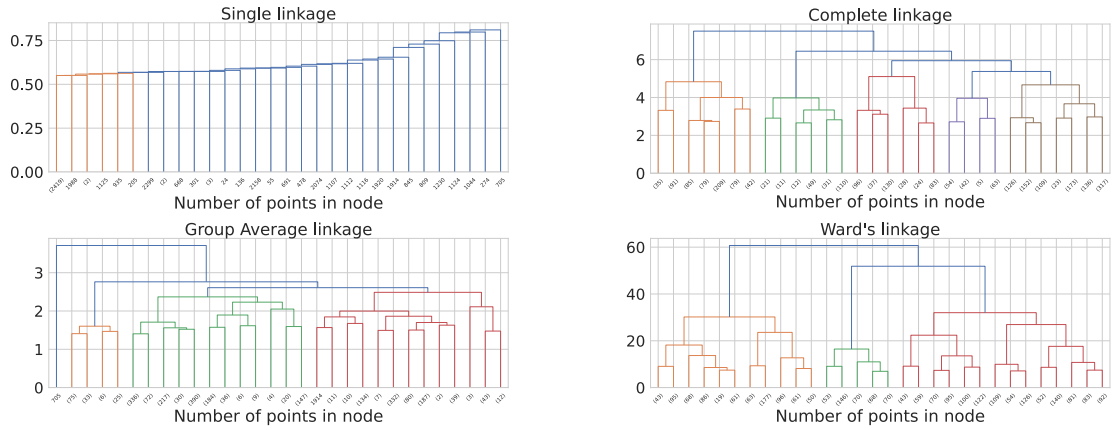

Figure 16: Comparison between dendrograms of linkage techniques with Agglomerative Clustering

interval. Merely based on a visual analysis of the dendrograms, the configuration with Ward's technique appears to lead to the formation of large clusters that are better distinguished and separated than the other types of linkage.

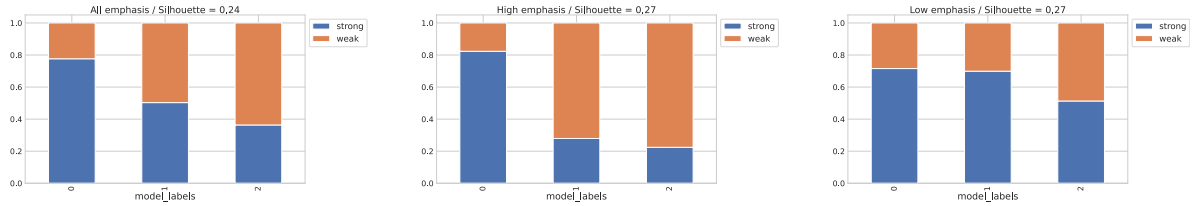

Figure 17: Emotion type clustering characterization with Agglomerative clustering (Ward)

Based on what was observed with the dendrograms, it was decided to test the agglomerative clustering algorithm using Ward's agglomeration technique, and by setting the number of clusters to 3. Starting from this configuration, the algorithm was tested on the 3 available datasets (complete, high emphasis, and low emphasis), analyzing which type of data favors a more accurate clustering of emotions. Looking at figure 17 it can be seen that, as already seen for most clustering algorithms used previously, the dataset containing high-emphasis data obtains visibly purer clusters than those obtained with the complete dataset: in the second graph, the first cluster is composed almost entirely of strong type emotions, while the second and third clusters are composed of weak type emotions.

3.5 Discussion of clustering results

After running and analyzing the behavior of various clustering algorithms, it can be seen that the Silhouette score and the SSE cannot always be considered the only criteria for determining the most performing algorithm.

Table 8: Comparison between Silhouette Scores of all tested clustering algorithms

Dataset	**KMeans**	**Bisecting**	**X-Means**	**DBSCAN**	**Agglomerative**
All emphasis	0.30	0.28	0.10	0.40	0.24
High emphasis	0.28	0.28	0.13	0.34	0.27
Low emphasis	0.29	0.28	0.12	0.41	0.27

Looking at table 8, it can be seen that the highest values are reached by the density-based algorithms (i.e., DBSCAN). However, as noted in section3.3, neither of the two algorithms tested resulted in clusters that were representative of the emotions or emotion types, and thus useful for our task. On the contrary, centroid-based algorithms (K-Means, Bisecting K-Means, and X-Means), and Agglomerative, have proven ability to identify clusters that were much more representative and characterizing of emotional states, although they tend to have a lower Silhouette score.

Looking again at the bar graphs for the emotional type, it can be seen that in almost all cases the high-emotional-emphasis dataset led the algorithms to generate visibly purer and more characterizing clusters, particularly in the case of the X-Means algorithm (see Figure 12). Agglomerative clustering, again using the high-emphasis dataset, also showed to generate clusters with an excellent degree of characterization, with a Silhouette score higher than X-Means, and closer to the values observed for K-Means and Bisecting K-Means.

In conclusion, the clustering analysis showed how a higher emotional emphasis allows for better clustering of individual emotions, and especially categories of emotions (emotion types).

4 Classification

In this section, some classification algorithms are tested: Decision-Tree (section 4.1), K-Nearest-Neighbors (section 4.2), and Naive-Bayes (section 4.3). All models were tested for the classification of the two variables on which the clustering algorithms were also tested: emotion and emotion type. Concerning the overall task of the project, all tests were performed for the dataset containing all data as well as for the datasets containing only data related to high or low emotional emphasis. The latter two datasets are about half the size of the original dataset, so it was decided to run tests also on a dataset containing both levels of emphasis, but of a size equal to 50% of the original one. In this way, a more accurate comparison can be conducted.

First, a test-set of the size of 20% of the original dataset was generated, which was used to test the classifiers trained with all 4 different datasets. In addition, the training-set obtained from the main dataset (containing data from both levels of emphasis) was used to perform hyperparameter tuning for each classification model. In the case of Decision-Tree and KNN, a Randomized-Search algorithm was chosen to speed up the search process. For Naive Bayes, given its low number of hyperparameters, a Grid-Search algorithm was opted to be used. In all three cases, a repeated Cross-Validation technique was exploited, configured with 20 splits and 3 repetitions. The number of splits was increased compared to the standard Scikit-Learn library to maximize the number of data with which each classifier is trained during the Cross-Validation iterations.

To conclude, an Accuracy baseline was also calculated for each dataset (for both variables under analysis). This gives a clearer idea of the performance of the models in comparison to a simplistic classifier. This was implemented using the *dummy classifier* from the Scikit-Learn library, configured through a stratified strategy.

4.1 Decision Tree

Before analyzing the Decision-Tree results on the test-set, a tuning step was performed on some of the hyperparameters of the model, specifically on:

- `criterion` (the function to measure the quality of a split), with *gini* and *entropy*;
- `max_depth` (the maximum depth of the tree), with several values from 2 to 100 (and the standard values of "None");
- `min_samples_split` (the minimum number of samples required to split an internal node), with several values from 2 to 10;
- `min_samples_leaf` (the minimum number of samples required to be at a leaf node), with several values from 1 to 10;
- `ccp_alpha` (complexity parameter used for Minimal Cost-Complexity Pruning), with several values from 0.0 to 0.05;

Tuning the hyperparameters showed that a configuration with `criterion` equal to "entropy," a `max_depth` equal to 18, and a `min_sample_split` value equal to 3 allowed for higher Accuracy. The best value for `min_samples_leaf` and `ccp_alpha` turned out to be respectively 1 and 0.0, as the default values of the *Scikit-Learn* library.

In the classification of individual emotions (see table 9), the results are different from what was hypothesized by observing the clustering results: the Decision-Tree trained only with the high-emphasis dataset obtains worse classification results than with the training-sets of the full dataset (*All*). The same behavior can also be observed in the binary classification of emotion types. In any case, the worst results

Table 9: Scores for emotions classification with Decision-Tree

Emphasis	Baseline	Accuracy	Precision	Recall	F1-Score	ROC AUC
Classification of emotions						
All	0.14	**0.33**	**0.32**	**0.33**	**0.33**	**0.62**
All (50%)	0.13	0.31	0.29	0.29	0.29	0.60
High	0.15	0.26	0.27	0.28	0.26	0.59
Low	0.15	0.19	0.20	0.19	0.19	0.54
Classification of emotion type						
All	0.50	**0.76**	**0.74**	**0.74**	**0.74**	**0.75**
All (50%)	0.52	0.71	0.69	0.69	0.69	0.69
High	0.49	0.66	0.64	0.65	0.64	0.65
Low	0.49	0.52	0.49	0.49	0.49	0.49

are obtained when the Decision-Tree is trained using only data with low emotional emphasis. This shows how a high emotional emphasis potentially allows emotions to be distinguished more correctly than a low emotional emphasis.

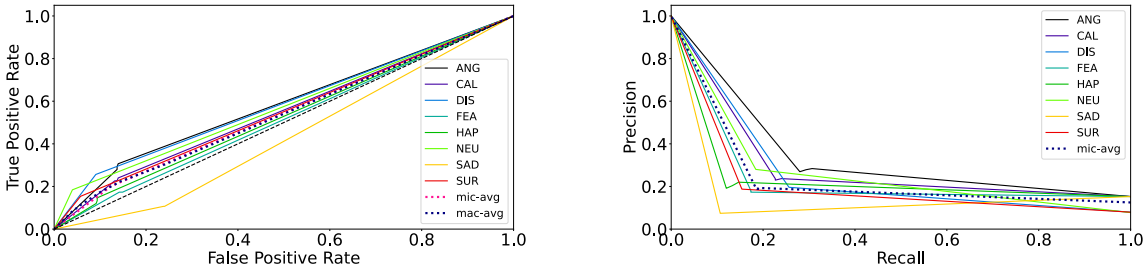

Figure 18: ROC and Precision-Recall curves for emotions classification with Decision-Tree

Figure 18 shows that there are some differences between the classification performances of every single emotion: emotions like *calm* and *anger* tend to obtain higher scores, compared with emotions like *sadness* or *happiness*. This behavior may indicate that some emotions, such as the last mentioned, could be more easily confused by a classifier. It is possible to assume that certain emotions share very similar acoustic characteristics and that their distinction should be traced to other types of factors.

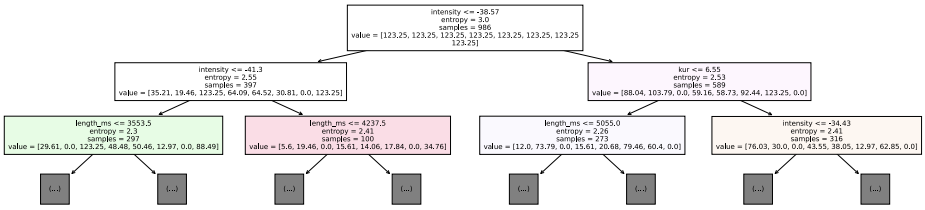

Figure 19: Partial Decision Tree trained on the complete dataset for emotions classification

Figure 19 shows the first 3 levels of the Decision-Tree generated with the training-set of the complete dataset. As can be seen, *length_ms*, *intensity*, and *kur* have a decisive weight in the early levels of the tree for the classification process. We can infer how intensity (understood as loudness), the length of the utterance, but also the kurtosis of the data distribution recorded by a specific utterance play a very important role in distinguishing individual emotions.

4.2 KNN

As already seen with Decision-Tree, also for KNN a tuning step was performed on the hyperparameters, specifically on:

- `metric` (metric to use for distance computation), set as *cityblock* or *euclidean*;
- `n_neighbors` (The number of neighbors indicated for each query), with all values between 1 and the length of the *training-set*;
- `weights` (weight function used in prediction), set as *distance* or *uniform*

After the tuning operations, all the parameters tested were changed compared to the Scikit-Learn library standards: the best `metric` was found to be *cityblock*, while the best function for the parameter `weights` proved to be *distance*. Instead, the optimal number for `n_neighbors` was found to be 3.

Table 10: Scores for emotions classification with KNN

Emphasis	Baseline	Accuracy	Precision	Recall	F1-Score	ROC AUC
Classification of emotions						
All	0.14	**0.42**	**0.41**	**0.42**	**0.42**	**0.76**
All (50%)	0.13	0.37	0.36	0.36	0.36	0.73
High	0.15	0.32	0.32	0.31	0.31	0.69
Low	0.15	0.30	0.30	0.29	0.29	0.67
Classification of emotion type						
All	0.50	**0.81**	**0.80**	**0.79**	**0.80**	**0.87**
All (50%)	0.52	0.75	0.74	0.73	0.73	0.78
High	0.49	0.71	0.70	0.71	0.70	0.76
Low	0.49	0.63	0.60	0.58	0.59	0.61

About the tests carried out by emphasis (see table 10), it can be seen that for both emotion and emotion type the best results are obtained on the full dataset, as already seen with Decision-Tree. However, it can be seen that even in this case a high emotional emphasis leads to better results than a low emotional emphasis, even if less than seen with Decision-Tree.

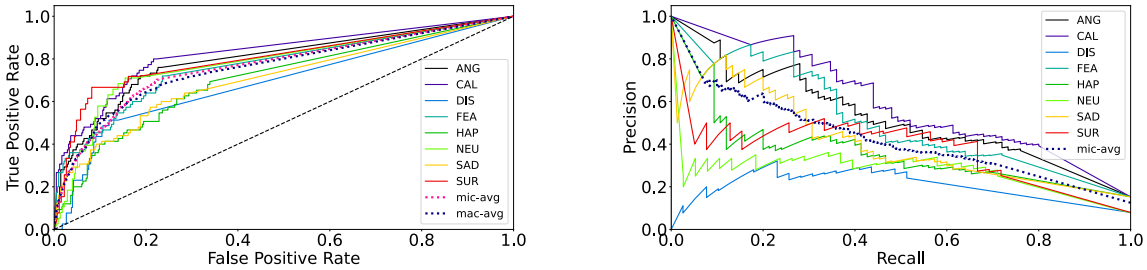

Figure 20: ROC and Precision-Recall curves for emotions classification with KNN

Overall, KNN achieves visibly better results than Decision-Tree. However, there is a more pronounced difference in performance between the individual emotions: image 20 shows that the best distinguishable and most easily classifiable emotions are *anger* and *calm*. In contrast, *sadness* and *happiness* remain more difficult to classify for KNN. *Disgust*, which Decision-Tree had been able to classify better than other emotions, turns out to be the emotion that performs the lowest with KNN (see the Precision-Recall curve). Thus, we can observe that KNN succeeds on average, but more heterogeneously across emotions. Decision-Tree, despite its lower performance, manages to provide more uniformity among the classification results of individual emotions, especially for emotions more difficult to distinguish during clustering, such as *disgust*.

4.3 Naive Bayes

As anticipated, for the Naive Bayes algorithm it was decided to tune the hyperparameter using GridSearch, specifically on `var_smoothing` (the portion of the largest variance of all features that is added to variances for calculation stability). It emerged that the best parameter for `var_smoothing` is a value equal to 5.37e-15.

As it can be seen from table 11, also in this case the results are different from what was assumed by observing the clustering results: the Naive Bayes trained only with the high-emphasis dataset obtains

Table 11: Scores for emotions classification with Naive-Bayes

Emphasis	Baseline	Accuracy	Precision	Recall	F1-Score	ROC AUC
Classification of emotions						
All	0.14	0.30	0.29	0.31	0.28	0.75
All (50%)	0.13	**0.35**	**0.34**	**0.35**	**0.33**	**0.75**
High	0.15	0.28	0.32	0.28	0.29	0.69
Low	0.15	0.23	0.21	0.23	0.21	0.63
Classification of emotion type						
All	0.50	0.75	0.74	0.74	0.74	0.81
All (50%)	0.52	**0.77**	**0.75**	**0.75**	**0.75**	**0.81**
High	0.49	0.67	0.65	0.66	0.66	0.72
Low	0.49	0.55	0.53	0.53	0.53	0.51

classification results almost equal to the Naive Bayes trained with the training-set of the full dataset. However, it is very interesting to note that a dataset reduced by 50% leads Naive Bayes to achieve visibly better results than with the full dataset. It is possible to assume that Naive Bayes does not particularly benefit from a larger dataset, in contrast to that seen with the other classifiers. On the contrary, a reduced dataset improves the overall performance of the model, for both variables.

This behavior slightly changes in the binary classification of emotion types (see table 11), where the Naive Bayes trained only with the high-emphasis dataset obtains visibly worse classification results than the one trained with the training-set from the full dataset.

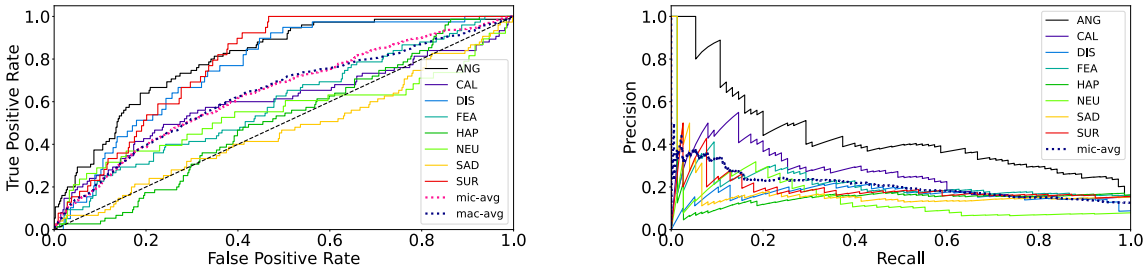

Figure 21: ROC and Precision-Recall curves for emotions classification with Naive Bayes

Also looking at the classification performance for each emotion (see figure 21), the results obtained seem similar to those of KNN. The results obtained for emotions like anger or calm are significantly better than those obtained for sadness or happiness. Neutrality, which other classifiers had been able to distinguish more accurately, was also very complex for Naive Bayes to classify.

4.4 Discussion of classification results

Using the complete dataset, all 3 models performed well in classifying emotion types, with Accuracy averaging 75/80%. In the classification of emotions, more marked differences are noted, with Decision-Tree and Naive-Bayes having an Accuracy near 30/35% while the KNN reaches an Accuracy of 42%. KNN emerges as the most accurate model, in all indicators used, and for both the attributes tested.

About the different levels of emotional emphasis, it can be seen that the classification results show a different situation from that observed in clustering. Whereas in an unsupervised approach the high emphasis allowed for a more characterizing clustering of emotions and emotion types, in the classification this does not happen. In all cases, using the full dataset, even if reduced by 50 percent, led to visibly better results than using only high or low-emphasis data. However, it can be seen that in all cases, using a high-emphasis dataset yielded better results than using a low-emphasis dataset.

Thus, we can summarize how higher or lower intensity plays a role in emotion recognition, and how it can ensure better performance than low emotional emphasis. However, to achieve excellent results, it is necessary to train classifiers to recognize emotions at both low and high emphasis. In these experiments, it was not possible to obtain excellent results using only high-emphasis data.

5 Filling Missing Values through Regression

In this section, various models for regression are compared: *Linear*, *Ridge*, *Lasso*, *Decision-Tree*, and *KNN*. The aim is to use the best regression model to fill `intensity` missing values in the dataset, and to test whether the performance of the classifiers improves.

Table 12: Regressor scores on test-set for *Intensity*

Regressor	**R^2-Score**	**Mean-Absolute-Error**	**Mean-Squared-Error**
Decision-Tree	0.68	3.22	18.07
KNN	0.75	2.79	13.02
Linear	**0.87**	**2.22**	8.05
Ridge	**0.87**	**2.22**	**8.04**
Lasso	0.86	2.23	**8.04**

The models were trained and tested only on the complete dataset (all emphasis) since during classification it was found to be the dataset that generally led to the best results. Also in the case of the regression models, a tuning of the hyperparameters was carried out, to obtain the best possible performance.[8] In addition, tests were performed while still splitting the dataset into training-set and test-set.

Table 12 shows how *Ridge* regressor provides the best results for each of the exploited indicators. Then it was decided to use this specific model to fill the missing values for *intensity*.

5.1 Retest of classifiers

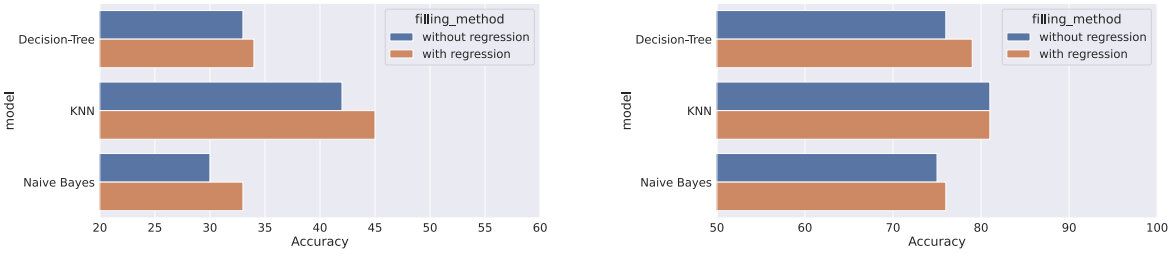

Figure 22: Comparison of classifiers Accuracy before and after regression (complete dataset)

Figure 22 shows how the filling of the missing values by regression leads to obtaining slightly better Accuracy results in classification. Increases occur on both variables tested (emotion and emotion type), although very slight and sometimes insignificant. Increases close to 3 percentage points can be detected in the classification of individual emotions with KNN and Naive Bayes, where 45% and 33% Accuracy on the full dataset are achieved, respectively. In all other cases, the increases are negligible, as in the case of emotion type classification with KNN, where no improvement occurred. In any case, it was decided to use this new dataset to proceed to the pattern mining phase.

6 Pattern Mining

This section discusses the results of pattern mining experiments. In the first part (section 6.1) the most relevant patterns and rules related to emotion types (strong and weak) were analyzed in detail, while in the second part the rules associated with individual emotions were analyzed. In the third and final part, the performance of a classification system based on the most relevant rules extracted in the previous sections were tested.

For pattern and rule extraction, the FP-growth algorithm was chosen, given its greater efficiency than the more classical Apriori algorithm. In addition, only certain attributes were selected from the original dataset: *intensity*, *length_ms*, *zero_crossings_sum*, *skew*, *mean* and *kur*. Therefore, only those attributes that enclose within them the average values of the more specific attributes were selected. In this way, it

[8] For simplicity, the detailed results and parameters of the hyperparameter tuning phase of each model have not been reported, as the regression section only constitutes an additional and non-fundamental experiment.

is simpler and more immediate to interpret and explain the rules identified. Attributes *Min*, *Max*, and *Std*, which showed to be highly correlated with each other and with *Intensity*, were excluded anyway.

Finally, each chosen attribute was discretized into several portions: 4 in the case of pattern mining for emotion types, and 8 in the case of pattern mining for individual emotions. The reason for this lies in the fact that in the case of emotion types we have 2 classes, while in the case of emotions these are 8, thus much more numerous.

6.1 Emotion type patterns and rules

Table 13: Discretization of selected attributes into quartiles

Attribute	Min.	Q1	Q2	Q3	Q4
Intensity	-63.87	-43.46	-37.303	-31.84	-15.58
length_ms	2936	3604	4004	4538	6373
Z.C.S.	4721	10362.50	12383.50	14966	30153
mean	-0.002	-1.39e-06	-9.81e-08	8.36e-07	0.001
kur	1.76	6.52	9.83	14.08	59.09
skew	-2.36	-0.34	0.004	0.26	1.80

As anticipated, for pattern mining related to emotion type, the numerical variables were discretized into 4 segments, illustrated within table 13.

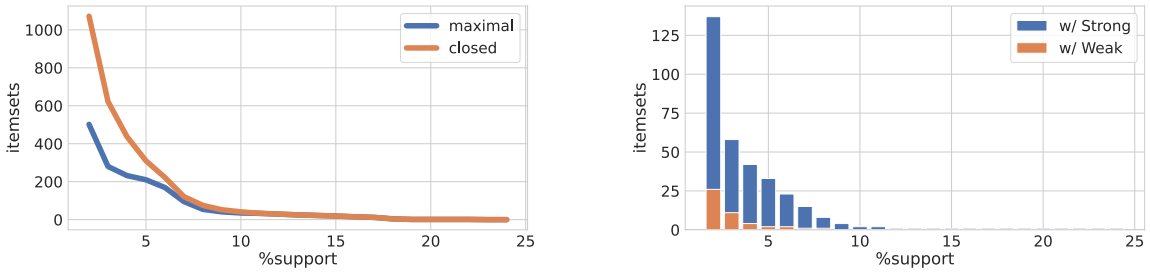

Figure 23: Decrease in the number of itemsets (left) and rules (right) as the percentage of minimum support increases

Next, it was observed how the number of itemsets (total) and rules (containing only "strong" or "weak") changed as the minimum support increased. Figure 23 shows that the number of itemsets and rules tends to stabilize with a minimum support value near 8 percent. However, it is interesting to note the number of rules related to the "weak" emotional type tends to stabilize at significantly lower support, about 4 percent. This fact may be a consequence of the lower number of items related to the weak type in comparison to the "strong" type, but also to a greater difficulty of the algorithms to identify non-overly specific rules for classifying weak emotions.

Table 14: Most relevant maximal patterns with emotion types

Itemset			Support	% Support
Strong	(-31.84, -15.58]_intensity		560	22.83
	(0.26, 1.80]_skew		463	18.88
	(14966, 30153]_zero_crossings_sum		451	18.39
	(-31.84, -15.58]_intensity	(14966, 30153]_zero_crossings_sum	264	10.77
Weak	(-63.87, -43.46]_intensity		411	16.76
	(1.76, 6.52]_kur		389	15.86
	(-2.36, -0.34]_skew		350	14.27
	(4538, 6373]_length_ms	(1.76, 6.52]_kur	269	10.97

Next, the 8 maximal patterns with the highest support (4 for "strong" and 4 for "weak") were analyzed: in table 14, it can be seen that the support rates of patterns related to "weak" are lower on average than "strong".

Analyzing the patterns of "strong" emotions, it can be seen immediately that these are associated with the fourth quartile of the attributes *intensity*, *skew* and *zero_crossings_sum*. In addition, the fourth quartile of intensity and the fourth quartile of zero_crossings_sum turn out to be the most frequent association (based on support) when maximal patterns with more than two elements are considered.

Concerning the "weak" type, it is possible to see that the situation for *intensity* and *skew* is exactly opposite to that seen for "strong": both attributes are found to be highly associated with the weak type in the first quartile. Among the most relevant patterns, the *zero_crossings_sum* attribute does not appear; on the contrary, weak emotions are associated with very low *kur* (kurtosis). The latter (observing the fourth "weak" pattern) also appears to be strongly associated with a very high value of *length_ms* (strong utterance length).

Table 15: Most relevant rules with emotion types

Consequent	Antecedents	Supp.	Conf.	Lift
Strong	(-31.84, -15.58]_intensity	24.92	91.65	1.49
	(0.263, 1.8]_skew	25.00	75.53	1.22
	(14966, 30153]_zero_crossings_sum	25.00	73.57	1.19
	(9.829, 14.085]_kur	25.00	72.76	1.18
Weak	(-63.87, -43.46]_intensity	25.00	67.05	1.75
	(1.76, 6.52]_kur	25.00	63.46	1.65
	(4538, 6373]_length_ms	24.06	58.81	1.53
	(-2.36, -0.34]_skew	25.00	57.10	1.49

For a more accurate analysis, it is necessary to extract rules from patterns, using more precise evaluation metrics, such as confidence and lift. In this case, it was decided to extract the rules with higher support, but only if they had a lift greater than 1. Three rules were extracted for each of the two emotional types.

As can be seen, rules related to "strong" and "weak" in table 15 coincide with the maximal patterns observed in table 14, for which the same considerations apply. In addition, it can be seen that "strong" emotions are associated with a medium-high (third quartile) kurtosis of the distribution (Kur).

In conclusion, it is possible to summarize what has been observed in the following points:

- As hypothesized, types of emotions can be effectively distinguished based on intensity: very high for "strong" emotions, very low for "weak" emotions
- Weak emotions are associated with very low kurtosis, which indicates a flatter distribution of values. This might indicate how the sound data for weak emotions are more varied, and fall in a wider range than the values for "strong" emotions, which are more concentrated and closer to the average
- High utterance length (*length_ms*) is associated with "weak" emotions, indicating that this type of emotion takes more time to be expressed
- The skewness (skew) of the distribution of "strong" and "weak" types are opposite: very high and very low respectively. This shows us that "strong" emotions tend to be associated and unified with higher values for most of the attributes in the dataset, while "weak" emotions at lower values

6.2 Individual emotions rules

To perform the individual emotions pattern mining task, the numerical variables were divided into 8 slices, which are shown in table 16.

In the case of emotions, for reasons of brevity, only two rules have been listed for each emotion following the same criterion used for emotion types.

Looking at table 17, it can be seen that the confidence values tended to be lower than those for emotion types, while the lift values tended to be higher. This indicates to us how the rules are even more strongly associated with and distinctive of each emotion than we have seen with the emotion types.

From an initial analysis, it can be seen that an intensity range is associated with almost every emotion, which confirms that the intensity value is very important to distinguish individual emotions. At the same time, however, it is observed that sometimes some emotions are associated with the same range of intensity. In these cases, relying on intensity alone is not enough to distinguish some emotions, especially if they belong to the same type.

Table 16: Discretization of selected attributes into 8 quantiles

Attribute	Min.	Q1/Q2	Q3/Q4	Q5/Q6	Q7/Q8
Intensity	-63.866	-47.36	-39.912	-34.925	-27.828
length_ms	2935.999	3437	3770	4271	4805
Z.C.S.	4720.999	9025	12383.5	17185.125	30153
mean	-0.001944	-4.27e-06	-4.27e-06	2.41e-07	0.00122
kur	1.757	6.52	14.085	17.801	59.086
skew	-2.358	-0.337	0.00426	0.263	1.8

Table 17: Most relevant rules with individual emotions

Consequent	Antecedents	Supp.	Conf.	Lift
Calm	(4805.0, 6373.0]_length_ms	12.23	40.00	2.60
	(-63.866, -47.36]_intensity	12.52	37.78	2.46
Angry	(-27.828, -15.578]_intensity	12.52	55.00	3.58
	(17185.125, 30153.0]_zero_crossings_sum	12.52	31.27	2.03
Disgust	(17.801, 59.086]_kur	12.52	17.58	2.24
	(-43.463, -39.912]_intensity	12.60	12.88	1.90
Sad	(4805.0, 6373.0]_length_ms	12.25	32.60	2.13
	(1.757, 4.94]_kur	12.52	25.73	1.68
Fearful	(-31.844, -27.828]_intensity	12.39	28.61	1.86
	(2935.999, 3437.0]_length_ms	12.88	20.88	1.36
Happy	(-31.844, -27.828]_intensity	12.39	27.30	1.78
	(0.263, 0.426]_skew	12.47	24.18	1.57
Neutral	(-63.866, -47.36]_intensity	12.52	17.26	2.25
	(1.757, 4.94]_kur	12.52	16.61	2.16
Surprised	(2935.999, 3437.0]_length_ms	12.88	27.21	3.47
	(-39.912, -37.303]_intensity	12.43	14.75	1.88

It is also interesting to note that for some emotions intensity is not the most important classification criterion; for sadness not even the second most important. In the case of sadness and calm, for example, very long utterance length is the most important criterion. Conversely, *surprise* is associated with a really short utterance length. it is also possible to note that disgust, an emotion that was more ambiguous than the others to distinguish, is associated with an average intensity, neither extremely high nor extremely low. On the whole, all the considerations made about patterns and the rules extended for types of emotions also apply here, to individual emotions.

6.3 Classification by rules

Following the analysis of the most frequent patterns and the most relevant rules, it was decided to implement classifiers for both variables under analysis (emotion and emotion_type). In the case of emotion_type, having only two classes (strong and weak) it was decided to exploit the rule with the highest lift among those detected in table 15. The algorithm classifies as "weak" all instances that respect that rule, and as "strong" all those that do not.

In the case of individual emotions, it was chosen to select the rule with the highest lift for each emotion from the table 17, excluding the emotion with the rule with the lowest lift. For each instance, the classifier checks the rules in order of lift (7 rules in all), but in case the instance does not comply with any of the rules, it will be classified with the emotion associated with the rule with the lowest lift (always referring to the table 17).

In both classification experiments, the results do not differ too much from those obtained using models such as Decision-Tree and Naive-Bayes. However, there is a substantive difference with the results obtained using KNN, which is the model that showed the best performance. In any case, the obtained percentages are a proof of how the extracted rules are relevant to the classification of emotions and their types. In our case, a rule-based model can obtain results not too far from those obtained with more advanced classification models.

Table 18: Main rule-based classification scores

Classification	Accuracy	Precision	Recall	F1-Score
Emotions	0.27	0.29	0.27	0.27
Emotion type	0.70	0.69	0.66	0.66

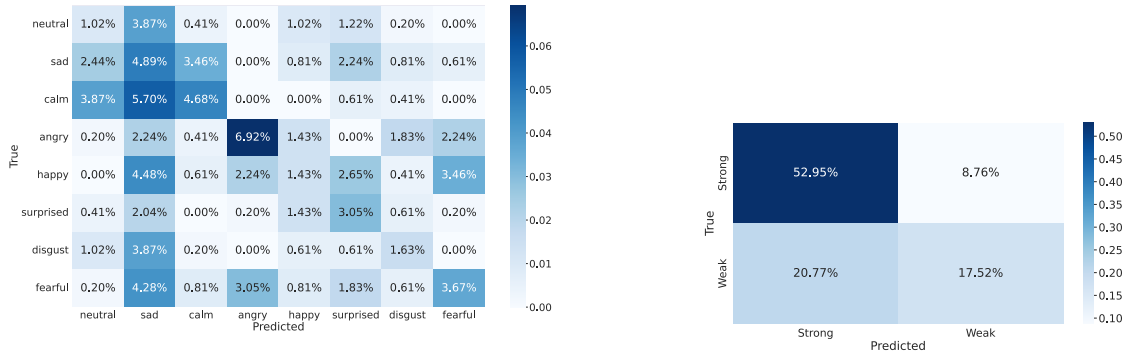

Figure 24: Confusion matrix for emotions (on the left) and emotion type (on the right) rule-based classification

Looking at the confusion matrices in Figure 24, it is also possible to analyze significant differences between the classification performance of each emotion and emotion type. Anger and calm, as already observed in most classification models, again turn out to be the easiest emotions to classify. In contrast, neutrality, disgust, and happiness turn out to be very complex to distinguish. Interestingly, frighten (*fearful*) is often classified by the model as happiness, probably due to the fact that they share the same range of Intensity. At the same time, the classifier tends to confuse calmness, sadness, and neutrality: again, the cause can be traced to the fact that the intensity and utterance length ranges (length_ms) are the same. Overall, it is possible to say that the attributes used are not fully sufficient to correctly classify all types of emotions.

7 Conclusions

In this report, we have gone through all the steps to extract information from a dataset, starting with understanding and preparing all the data (chapter 2). In chapter 3, we looked at clustering and at various algorithms, after which we moved on to classification with three main models (section 4). After trying to fill in the missing values using regression (section 5, further classification tests were performed in search of improvements. Finally, numerous pattern mining operations were performed (section 6), looking for frequent rules and patterns most associated with individual emotions and their types.

During clustering, it was possible to see that the high-emphasis dataset allowed for more characterizing clusters of emotions and emotion types. In contrast, classification showed that a full dataset of both levels of emphasis was necessary to obtain higher results. In any case, the high-emphasis dataset yielded visibly better results than the low-emphasis dataset. In summary, to answer the question used as the title of this project, it is possible to state how intensity is indeed important in emphasizing emotions, and in enhancing their distinction and identification. In a context such as supervised learning, however, it is important that the models have at their disposal a more varied and complete dataset, including multiple levels of emphasis.

Pattern mining showed that for recognizing most emotions the intensity parameter is very important: for almost all emotions there is a rule associating a certain intensity range. However, it has been seen that intensity alone is not sufficient to enable accurate classification: for several emotions, other attributes must also be considered.

In conclusion, Intensity is probably not sufficient to comprehensively define a high or low level of emphasis. In other words, it is not always sufficient to raise or lower one's voice to better express an emotion: other factors must be taken into account.