NOVA IMS Information Management School

MGI

Mestrado em Gestão de Informação
Master Program in Information Management

# THE ROLE OF ARTIST AND GENRE ON MUSIC EMOTION RECOGNITION

Pedro Miguel Fernandes Vale

Dissertation presented as partial requirement for obtaining the Master's degree in Information Management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

**NOVA Information Management School**
**Instituto Superior de Estatística e Gestão de Informação**
Universidade Nova de Lisboa

# THE ROLE OF ARTIST AND GENRE ON MUSIC EMOTION RECOGNITION

by

Pedro Miguel Fernandes Vale

Dissertation presented as partial requirement for obtaining the Master's degree in Information Management, Specialization in Knowledge Management and Business Intelligence.

**Supervisor:** Prof. Dr. Rui Pedro Paiva

August 2017

# ABSTRACT

The goal of this study is to classify a dataset of songs according to their emotion and to understand the impact that the artist and genre have on the accuracy of the classification model. This will help market players such as Spotify and Apple Music to retrieve useful songs in the right context.

This analysis was performed by extracting audio and non-audio features from the DEAM dataset and classifying them. The correlation between artist, song genre and other audio features was also analyzed. Furthermore, the classification performance of different machine learning algorithms was evaluated and compared, e.g., Support Vector Machines (SVM), Decision Trees, Naive Bayes and K-Nearest Neighbors.

We found that Support Vector Machines attained the highest performance when using either only Audio features or a combination of Audio Features and Genre. Namely, an F-measure of 0.46 and 0.45 was achieved, respectively. We concluded that the Artist variable was not impactful to the emotion of the songs.

Therefore, by using Support Vector Machines with the combination of Audio and Genre variables, we analyzed the results and created a dashboard to visualize the incorrectly classified songs.

This information helped to understand if these variables are useful to improve the emotion classification model developed and what were the relationships between them and other audio and non-audio features.



# KEYWORDS

Music Emotion Recognition (MER); Music Information Retrieval (MIR); Songs; Emotions; Data Mining; Classification; Support Vector Machines

# Table of Contents

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF EQUATIONS

# ACRONYMS

| Term | Definition |
|---|---|
| BPM | Beats per minute |
| DEAM | Database for Emotional Analysis in Music |
| GEMS | Geneva Emotional Music Scale |
| KNN | K-Nearest Neighbours |
| MER | Music Emotion Recognition |
| MFCC | Mel-Frequency Cepstral Coefficient |
| MIR | Music Information Retrieval |
| MIREX | Music Informational Retrieval Evaluation eXchange |
| NA | Negative Affect |
| NB | Naive Bayes |
| PA | Positive Affect |
| RBF | Radial Basis Function |
| RMS | Root-Mean-Square Energy |
| SFM | Spectral Flatness Measure |
| SVM | Support Vector Machines |

# 1. INTRODUCTION

Music is an important part of one's individual and collective culture. It is used to carry emotions from the composer to the listener (Hevner, 1935).

Nowadays, music is a colossal industry with global revenues of 28.7 billions of dollars in 2015, a growth of 11.6% compared with the former fiscal year (Ellis-Petersen, 2016). Spotify and Apple offer consumers a new way to listen to music: music streaming. In 2015, these streamers observed a 66% growth in subscriptions and helped the digital sales outperform the physical sales for the first time (Ellis-Petersen, 2016). These companies created enormous databases of songs, listeners and their preferences, and now have the need to retrieve the right song at the right time to the right listener.

## 1.1. Background and Problem Identification

The digital era brought numerous opportunities to the music business, expanding exponentially the number of data stored on the listener's behaviors and tastes. Music streamers can now collect data while we listen to music in our car, while working, exercising, and during other activities by using their services.

The growth led to every person having a considerable collection of albums and songs. With the opportunities arrived the limitations of music cataloging and retrieval, which is still a relatively new area of research. The problem is retrieving relevant songs in a given context from gigantic databases (Panda, 2010).

This is the problem that the Music Information Retrieval (MIR) field of research tries to solve. There is a great effort in analyzing similar listeners to suggest new songs (Hu & Liu, 2010) but MIR researchers go a step further and analyze audio, melodic features and lyrics (Panda, Rocha, & Paiva, 2015).

Music Emotion Recognition (MER) is a relatively new field which investigates what emotions each song carries on to the listener. One of the problems is the non-existence of a universal definition of Emotion. Psychologists have made efforts to create frameworks in which we could classify emotions, but there is not a consensus yet (Kleinginna & Kleinginna, 1981).

Although researchers are working with not consensual data, MER has gained interest and nowadays there are researches on multiple features of music such as Timbre, Tempo, Lyrics and Genre in order to understand which emotion is represented in the songs (Panda, 2010).

Spotify, Pandora and other major players on the music industry are now sponsoring MIR and MER events held by the International Society of Music Information Retrieval. Spotify's Weekly Discover Playlists, which suggest to listeners songs related to the ones they listened in the pasted by using emotion classification models and user-related metrics, had 1.7 billion streams in the first six months since it was created (Pasick, 2015).

One feature that have not been studied extensively was the Artist that created the song. Some artists have the habit of creating uplifting songs, while others prefer sad songs. The relationship between Artist, Genre and other audio and melodic features are an important step towards understanding which Emotion is expressed in the song.

## 1.2. Objectives and Approaches

The main objective is to analyze what is the impact that Artist and Genre have on the Music Emotion Recognition model when combined with other audio features. By creating multiple classification models with audio features and non-audio features, we aim to offer a contribution to understanding what is the role of the Artist and Genre variables on the overall classification model.

Secondly, we will analyze the correlation between Audio variables, Genre and Artist to observe which are the relationships between these song features.

Thirdly, even though there have been researches pointing that Support Vector Machines (SVM) outperform other Machine Learning algorithms, we will compare SVM with Decision Trees, Naive Bayes and K-Nearest Neighbors (Laurier & Herrera, 2007).

Lastly, we consider important the creation of a dashboard that helps to observe the misclassified songs to understand where the main weaknesses of the model are, especially concerning genres and artists.

# 2. LITERATURE REVIEW

## 2.1. Music and Emotion

Since the main goal of this research is to study the role of emotions in music, it is important to have a clear definition of emotions and which are the main differences between the former and moods.

Kleinginna and Kleinginna (1981) studied the definitions of emotion in order to find a universal terminology. They came up with the following definition: "Emotion is a complex set of interactions among subjective and objective factors, mediated by neural/hormonal systems, which can (a) give rise to affective experiences such as feelings of arousal, pleasure/displeasure; (b) generate cognitive processes such as perceptually relevant effects, appraisals, labelling processes; (c) activate widespread physiological adjustments to the arousing conditions; and (d) lead to behaviour that is often, but not always, expressive, goal-oriented, and adaptive." (Kleinginna & Kleinginna, 1981).

The American Oxford Dictionary[1] defines emotions as "A strong feeling deriving from one's circumstances, mood, or relationships with others.". This means that one of the factors that influences emotions are moods. The definition of mood from the American Oxford Dictionary is "a temporary state of mind and feeling".

One person can be sad for days, having this mood without any apparent reason. This mood will influence the appearance of emotions like angriness and sadness. "The concept of mood is complex and difficult to establish. It reflects a moving notion that cannot be easily grasped. (...) The conception of mood in cognitive psychology is derived from the analysis of emotion. While emotion is an instantaneous perception of a feeling, mood is considered as a group of persisting feelings associated with evaluative and cognitive states which influence all the future evaluations, feelings and actions" (Amado-Boccara, Donnet, & Olié, 1993). As stated, moods are difficult to define just like emotions.

It is important to differentiate these two concepts since both have a similar meaning. This difference can help explain their own definition:

- Duration: Emotions have a shorter duration when compared to moods;
- Intensity: Emotions are more intense than moods;
- Cause: Emotions can be aroused from moods while mood causes are usually unknown;
- Strength: Emotions are stronger than moods.

[1] https://www.oxforddictionaries.com/

When one watches his favorite movie or listens to his favorite song, an emotion will arise from the joy (mood). In this example, the cause was joy and the emotion happiness arouse from it. Emotions are stronger than moods because, when we are in a bad mood, we can still have moments of happiness. Emotions and moods are usually used interchangeably on the MIR research but we will opt to use the term Emotions, as we consider that it is the most accurate in the context of MER, where short music clips are considered.

Furthermore, in MER, emotions are commonly divided into three categories: expressed emotions, perceived emotions and induced emotions (Gabrielsson, 2001):

- Expressed emotion indicates the emotion that the performer wants to share with the listeners (Gabrielsson & Juslin, 1996);
- Perceived emotion refers to the emotions the listeners apprehend as being present in the song, which is not always equal to the expressed emotion or the emotion felt by the listener (Gabrielsson & Juslin, 1996);
- Induced emotion regards the emotion the listener feels in response to a song (Scherer & Zentner, 2001), i.e., the emotion felt while listening to the song.

Wager et al. determined that perception and induction of emotions are associated with ´peak activations' in different areas of the brain, demonstrating that these are two distinct processes (Wager et al., 2008). These two categories of emotions do not have always a positive relation. There can also be negative, no systematic relation or no relation at all (Gabrielsson, 2001). MIR researches tend to focus on the perceived emotion, mainly because it is less dependable to situational factors (Y.-H. Yang & Chen, 2012). One performer might attempt to create a music that will transmit sadness, but the listener might perceive calmness, despite actually making him feel happier. We will focus on perceived emotions in this research.

Since emotions and moods are subjective, varying from person to person and also across cultures, there is not a standard emotion taxonomy. The two main approaches are categorical models and dimensional models. We will now dive into the more notorious emotion paradigms.

## 2.2. Emotion Paradigms

In the literature, emotion models fall into two main paradigms: the dimensional and the categorical paradigms. The main difference between them is the usage of discrete categories to represent emotions in the categorical models, while dimensional models represent emotions along 2 or 3 axes as discrete adjectives or as continuous values (Kim et al., 2010; Russell, 1980).

### 2.2.1. Categorical Paradigm

In this paradigm, two of the first and most well-known models are Hevner's and Ekman's.

Ekman defended the existence of six categories, named 'basic emotions', which are the basis for the non-basic emotions (Ekman, 1992). The basic emotions, anger, disgust, fear, happiness, sadness and surprise, were developed according to facial expressions, which means that some of them (e.g., disgust) do not suit music well. There is also the issue of not having moods associated to music (e.g., calm) (Hu & Downie, 2010).

This representation is questioned by Ortony et al. (1990), who compare the notion of 'basic emotions' to 'basic natural languages'. According to them, there are not basic natural languages, what exists is a great diversity of natural languages depending of the culture and other factors (Ortony & Turner, 1990).

Kate Hevner is one of the first researchers of music psychology. She concluded that music always carries emotional meaning and that this meaning is not entirely subjective, so it can have some reason to it. Hevner's model consists of eight clusters using a total of 67 adjectives (emotions) organized in a circular way (Figure 1). Inside each cluster, adjectives have a close meaning (intra-cluster similarity). Distant clusters are less similar than adjacent clusters (e.g., the adjectives joyous (cluster 6) and humorous (cluster 5) have more similarities than the adjective joyous (cluster 6) and dreamy (cluster 3)) (Hevner, 1935).

Farnsworth (1954) adapted Hevner's model and the result was nine adjective groups (Farnsworth, 1954).

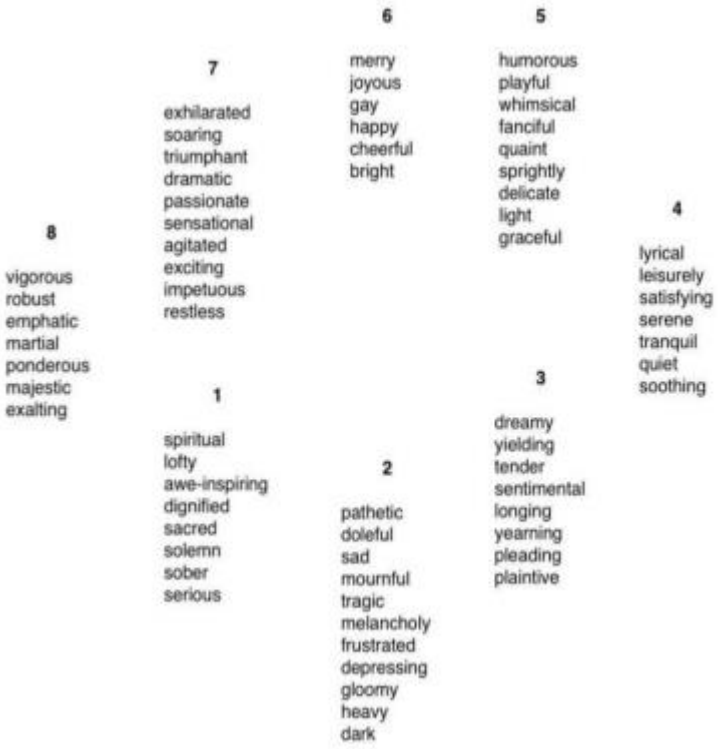



*Figure 1 - Hevner's Model (Hevner, 1935)*

MIREX (Music Information Retrieval Evaluation eXchange) is a framework used by the Music Information Retrieval (MIR) scientific community for the evaluation of systems and algorithms (Downie, 2008). This framework classifies songs into one of five clusters shown in Table 1.

| **Clusters** | **Mood Adjectives** |
|---|---|
| Cluster 1 | Passionate, Rousing, Confident, Boisterous, Rowdy |
| Cluster 2 | Rollicking, Cheerful, Fun, Sweet, Amiable/Good Natured |
| Cluster 3 | Literate, Poignant, Wistful, Bittersweet, Autumnal, Brooding |
| Cluster 4 | Humorous, Silly, Campy, Quirky, Whimsical, Witty, Wry |
| Cluster 5 | Aggressive, Fiery, Tense/anxious, Intense, Volatile, Visceral |

*Table 1 - The five MIREX clusters and respective subcategories (Malheiro, 2016)*

According to Laurier et al., clusters 2 and 4 overlap semantically, and there is an acoustic overlap between clusters 1 and 5 (Laurier, Grivolla, & Herrera, 2008). Another major issue is that the emotions used in the MIREX model cannot represent the entire universe of emotions perceived by humans when listening to music (Y.-H. Yang & Chen, 2012).

To address the specific needs of induced emotions, which none of the mentioned above models did, Zentner et al. (2008) developed a domain-specific scale named Geneva Emotional Music Scale (GEMS). Zentner et al. conducted four studies: the objective of the first and second studies was to create a list of terms for both perceived and felt emotions in which the conclusion was that the five groups of listeners evaluated showed a considerable variability across musical genre and perceived vs induced emotions. The difference was higher on the positive emotions in induced emotions. “As people move into a mental state in which self-interest and threats from the real world are no longer relevant, negative emotions lose their scope.” This remark by Zentner et al. is the justification on why induced emotions tend to be more positive. On the third and fourth studies, the researchers used factory analysis of questionnaire data of music-induced emotions to create the GEMS scale. Further factor-analysis added shorter versions of the 45 terms of the GEMS scale. The 9-term scale, which is one of the shorter versions, include the following terms: Wonder, Tenderness, Transcendence, Nostalgia, Power, Peacefulness, Joyful, Tension, Activation and Sadness (Zentner, Grandjean, & Scherer, 2008).

Coutinho & Scherer (2012), confirmed the structure of GEMS with an experiment in which the problem of overrepresentation of classical music on the original research was addressed. New terms were suggested related to feelings of harmony, interest and boredom (Coutinho & Scherer, 2012).

Aljanaki (2016) points out that the conclusion in the original work, in which it is showed that the GEMS scale is a more precise instrument to measure musical emotion than Valence-Arousal or basic emotions, can be questioned. The main reasons are the small size of the experiment, the overrepresentation of one genre and the unconventionality of the questions regarding the Valence-Arousal model (Aljanaki, 2016).

All in all, the major limitation of the categorical models is the ambiguity present on the proposed taxonomies, since there is no definitive way to discriminate the adjectives, especially the ones that are closer in meaning (Y.-H. Yang, Lin, Su, & Chen, 2008).The result is the possibility of having high intra-cluster heterogeneity (Y.-H. Yang & Chen, 2012).

### 2.2.2. Dimensional Paradigm

In this paradigm, the authors propose the use of a multi-dimensional space in order to plot the locations of the emotions. Commonly, a two-dimension approach is adopted, comprising two axes: arousal against valence. The Y-axis represents arousal (also known as energy, activation and stimulation level), and the X-axis represents valence (polarity of the emotion, also known as pleasantness, either positive or negative). This creates a four-quadrant interpretation corresponding to different emotions (Russell, 1980), as illustrated in Figure 2.

There is also a variation which includes a third dimension: dominance or potency, in order to differentiate between close emotions (e.g., fear (negative dominance) and anger (positive dominance) have similar arousal and valence values and, hence, might be distinguished by dominance) (Tellegen & Clark, 1999). For the sake of simplicity and visualization, MER works usually do not apply this dimension.

Dimensional models can be further divided into discrete and continuous models. Discrete models address the representation of the emotion on one of the four quadrants, each having more than just one emotion (e.g., happiness and surprise are both high arousal and valence, belonging in the first quadrant). Continuous models approach each point on the continuous space as one emotion, reducing the ambiguity present on the discrete models. The two most well-known models are Russel's Model and Tellegen-Watson-Clark's Model.

Russel's can be considered as both discrete and continuous model, because it is composed by four categories, one for each quadrant (happy in quadrant 1, angry in quadrant 2, sad in quadrant 3 and relaxed in 4). Furthermore, Russel's Model also represents a list of of 28 adjectives that are situated in the Cartesian plane (Figure 2)**.** Emotions are typically far from the center where arousal and valence have small values, representing unclear and unidentifiable emotions (Russell, 1980).


*Figure 2 - Russell's Model of Emotion (Calder, Lawrence, & Young, 2001)*

Tellegen-Watson-Clark's Model is composed of a third dimension, and follows an innovative hierarchical perspective. A three-level hierarchy composed on the highest level by pleasantness vs un-pleasantness, an independent positive affect (PA) versus negative affect (NA) dimension at the second level, and discrete expressivity factors of joy, sadness, hostility, guilt/shame, fear emotions at the base level (Figure 3) (Tellegen & Clark, 1999).

Despite the lack of differentiation of close points (emotions) on the valence-arousal plan, it compensates in simplicity when compared with the third-dimensional models.

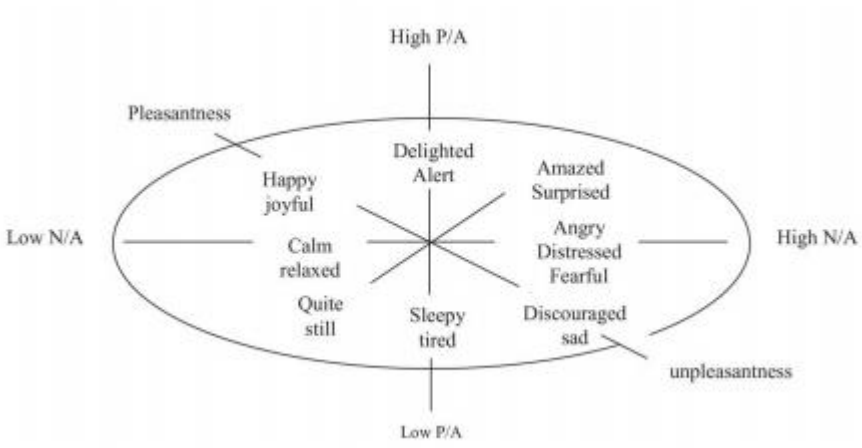



*Figure 3 - Tellegen-Watson-Clark Model of Emotion (Trohidis, Tsoumakas, Kalliris, & Vlahavas, 2011)*

A brief summary of the models discussed above is presented in Table 2:

| Emotion Model | Type | Granularity (cluster/emotions) |
|---|---|---|
| (Hevner, 1936) | Categorical | 8/67 |
| (Farnsworth, 1954) | Categorical | 9 |
| MIREX | Categorical | 5/29 |
| GEMS | Categorical | 9/40 |
| (Russell, 1980) | Dimensional | 4 (discrete view) or 28 (continuous) |
| (Tellegen et al., 1999) | Dimensional | ∞ |

*Table 2 - Comparison of emotion models*

## 2.3. General MER Review

### 2.3.1. Research Review

Whilst being a relatively new research field, Music Emotion Recognition (MER) gained interest from researchers of diverse backgrounds.

The first paper on emotion detection in audio, to the best of our knowledge, was published by Feng et.al (2003). A system for emotion detection in music using only features of tempo and articulation was proposed to identify emotions in a music piece. The categorical classification model included only four emotions: happiness, anger, sadness and fear and it was performed using a neural network with three layers. The accuracy of the model of the happiness, sadness and anger categories was considerable, between 75% and 86%. On the other hand, the observed accuracy of the fear category was only of 25% (Feng, Zhuang, & Pan, 2003). The main issue with this research is that the sample is too small, with a collection of 330 songs for training and only 23 for testing, which is less than 7% of the total. Another problem can arise from the lack of representation of the fear category on the test data, with only 3 songs. There is also an absence of relevant information regarding the annotation process details and about the musical genres.

Li et al. (2003) were one of the first researchers to address MER as a multi-label classification problem by analyzing music signals. Multi-label classification regards songs as having the possibility to being associated with more than one emotion. As stated in the paper “In emotion detection in music, however, the disjointness of the labels is no longer valid, in the sense that a single music sound may be classified into multiple emotional categories.” The process was divided into feature extraction and multi-label classification. The authors used the Farnsworth adjective groups, which is a categorical

model, and added three additional groups: mysterious/spooky, passionate and bluesy. This adjective groups were grouped into six supergroups. The dataset consisted of 499 sound files extracted from 128 albums. Thirty audio features were extracted using the Marsyas software framework, belonging to three different categories: timbral texture features, rhythmic content features, and pitch content features. In order to build the classifiers, the authors used SVM. The dataset was divided into 50% training data and 50% test data, resulting in an F-measure micro average of 44.9% and a 40.6% macro average. The main problems with the article are within the dataset: the songs dataset was labeled by a single subject. In addition, only some genres were represented on the dataset: Ambient (120 files), Classical (164 files), Fusion (135 files), and Jazz (100 files) (Li & Ogihara, 2003).

Later that year, Liu et al. (2003) used a hierarchical framework against a non-hierarchical framework to classify acoustic music data using audio features: intensity, timbre and rhythm. The authors decided to use Thayer's model of mood as the emotion framework and the 800 twenty seconds long music clips were labeled by three music experts. The results ranged from 76.6% to 94.5% for the hierarchical framework and from 64.7% to 94.2% for the non-hierarchical framework (D. Liu, Lu, & Zhang, 2003). The main limitation of this study is the fact that the scope of the classifier is restricted to classical acoustic music.

Aiming to increase efficiency in the annotation process, Yang et al. (2004), used a dataset of 152 Alternative Rock music samples of 30 seconds each, of which 145 had lyrics, to extract emotion intensity information. The focus of the research were the negative emotions, since the authors considered them to be more ambiguous. The Tellegen-Watson-Clark's mood model was used. The correlation between emotion intensity and rhythm and timbre features was 0.9. In order to separate like-valenced emotions, lyrics were used and observed an accuracy of 82.8% (D. Yang & Lee, 2004).

Yang et al. (2008) tried to predict arousal and valence values, using Russel's model of mood. The dataset consisted of 195 songs from Western, Japanese and Chinese albums, which were distributed uniformly in each quadrant of the emotion plane. The authors used Support Vector Regressions to estimate valence and arousal values, employing features such as pitch, timbral texture and pitch content. The R2 statistic observed was 58.3% for arousal and 28.1% for valence. Even though the authors demonstrated with Principal Component Analysis that valence and arousal were not truly dependent, the residual dependence can be deteriorating the accuracy of the model (Y.-H. Yang et al., 2008).

When comparing multi-label algorithms and features on the Tellegen-Watson-Clark's mood model, Trohidis et al. (2011) observed that the RAKEL algorithm had the best accuracy when predicting

the emotion from 100 songs belonging to 7 different genres: Classical, Reggae, Rock, Pop, Hip-Hop, Techno and Jazz (Trohidis et al., 2011).

Bischoff et al. (2009), incorporated audio features with social annotations from Last.fm[2] and compared the accuracy of the model against labels from AllMusic[3], which are annotated by music experts. The 4737 song' dataset was evaluated using 3 different models: audio-based using SVM, tag-based using Naive Bayes and a linear combination of the former. The emotions were predicted according to MIREX, Thayer's mood models and theme's manual clusters from AllMusic. Audio features included BPM, MFCCs and spectral centroid. The highest accuracy was found in the theme's clusters under the linear combination of social tags and audio features with an average accuracy of 62.5%. The labels used from Last.fm can have accuracy issues since any user can create these tags and associate them with a song. Likewise, as stated by the authors, the mood-related labels are less frequent on the platform, where users usually give more attention to the creation of genre-related labels (Bischoff et al., 2009). Also, the annotation process in AllMusic is unclear, namely, it is unclear if the annotators are focused on the audio part, lyrics part or both.

Recently, Song et al. (2012), studied which musical features are more relevant for emotion classification. The analysis of 2904 pop songs was conducted using SVM classifiers associated with two kernels: polynomial and radial basis functions. The results were trained and evaluated against Last.fm tags and audio data from 7digital[4]. The authors used a recent categorical emotional model named Geneva Emotion Musical Scale (GEMS). This study found that, by combining spectral, rhythmic and harmonic features, the model presented the best accuracy of 54% (Song, Dixon, & Pearce, 2012). One limitation of this work, in the context of our research, is that the GEMS model aims at induced emotions rather than perceived emotions.

Other features besides audio started being used in MIR and MER. Laurier et al. (2008) combined Natural Language Processing techniques on lyrics and MIR techniques on audio features. The assumption of this study was that part of the semantic information were present solely on the lyrics, hence it will be beneficial to add lyrics to the classification model. The research was divided into 3 models: audio-only, text-only and a bi-modal classification system integration of the two former models. The authors used a categorical emotion model with the following categories: happy, sad, angry and relaxed, corresponding to each of the four Russell's quadrants. Each category is binary, so one

[2] https://www.last.fm
[3] http://www.allmusic.com/
[4] http://www.7digital.com/

song can be either 'sad' or 'not sad'. The dataset is a collection of mainstream popular music and the selection was made using Last.fm tags and lyrics from LyricWiki[5]. Lastly, the listeners were asked for validation of the song's emotion, and only songs that were validated by at least one listener were kept on the dataset. The final dataset was composed of 1000 songs. In order to classify the songs, three algorithms were compared: SVM, Logistic Regression and Random Forest. SVM performed better than the rest. The audio features used for the classification model were: timbral (e.g., spectral centroid), rhythmic (e.g., tempo), tonal (Harmonic Pitch Class Profiles) and temporal descriptors. The classification model with the best accuracy was the combination of audio and lyrics, showing considerably higher accuracy, reaching the peak difference around 5 percentage points in the categories 'sad' and 'happy', when compared with the audio-only and lyrics-only model. The authors mitigated the risk of error on the Last.fm tags by using the listener's manual validation. Most studies referred above which used Last.fm as a source for social tags, did not use this methodology. Despite this, the annotations were only made by listening to 30 seconds of each song, creating the possibility of a biased analysis (Laurier et al., 2008).

In another study using lyrics as a feature of the emotion classification model, Hu et al. (2010) used a dataset of 5,296 songs to study lyrics versus audio-based models. Russel's emotion model was selected and the songs were divided into 18 mood categories according to the tags generated by Last.fm users. Each mood categories were treated with a binary approach. From the 18 mood categories, 7 divergent categories have shown lyrics outperforming audio features and only 1 category where audio features had a better performance. It was also demonstrated that the lyrics have a strong semantic connection with the mood categories (Hu & Downie, 2010). The main problem with this study is that the 'ground truth' used was solely Lasfm.com tags and reviewed by two experts, which could be biased.

MIDI has been shown to be a good choice in order to have impact on the accuracy of emotion classification models, in particular on the valence dimension (Yi Lin, Chen, & Yang, 2013). Lu et al. (2010), extracted features from MIDI files such as Duration, Voice Separation, Acoustic Guitar Friction and Average Melodic Interval, to compare with audio feature and lyrics-based models. The authors used Thayer's mood model and Adaboost to train a classifier that combined audio, lyrics and MIDI features. The dataset consisted of 500 Chinese pop songs labeled by 8 participants. The observation of the accuracy comparison between MIDI-only, lyrics-only, audio-only and the combined model shows that the combination of features is the most accurate model (72.4%). In the comparison results, it is

[5] http://www.lyricwiki.com/

shown that the combination between audio and lyrics has an accuracy of 72%, which is only 0.4 percentage points lower than the best model, which is the combination of lyrics, audio and MIDI. This can hint that there is redundancy in the information between audio and MIDI features. Secondly, the dataset was comprised solely of Chinese pop songs, which makes it difficult to compare this paper's model with other models developed mainly for Western songs (Lu, Chen, Yang, & Wang, 2010).

Besides standard audio features, lyrics and MIDI features, recently Panda et al. (2015) developed a classification model with Standard and melodic audio features. The authors built the dataset extracted from AllMusic API to be close to MIREX Mood Classification Task, a categorical model which has five clusters. The dataset consisted of 903 audio-clips of 30 seconds each. Melodic features extracted included pitch and duration features, vibrato features and contour typology. The best score resulted from using only 11 features with a F-measure of 64%. The conclusion was that the best results appear to be when there was a combination of melodic and standard audio features (Panda et al., 2015).

The best score obtained in the MIREX Audio Music Mood Classification Task, which is a competition held every year where researchers classify a dataset unknown to contestants, was observed in 2011 with a classification accuracy of 69.5%. We can compare different years because the dataset remained the same since the inception of the contest[6].

### 2.3.2. Artist and Genre Review

In the artist and genre field, there is not a considerable amount of research. We will dive deeper into the three most well-known studies in this specific part of MER.

**Exploiting Genre for Music Emotion Classification** (Yu-ching Lin, Yang, Chen, Liao, & Ho, 2009)

This research uses a two-layer emotion classification scheme: in the first-layer the genre classification model takes place and in the second-layer the authors apply the genre-specific emotion classification model.

The dataset is composed of 1535 songs collected from 300 albums and enriched with information from the review website AllMusic: 12 emotions and 6 genres. Each of the albums can have more than one emotion but can only have a single associated genre. Songs are converted to 22050Hz,

[6] http://www.music-ir.org/

mono channel PCM WAV and feature extraction was done using Marsyas. The software extracted 436 audio features.

The authors prove through the Chi-Squared test of association that genre and emotion are not independent.

Firstly, a model of genre classification was created. Afterwards, on the second layer, one emotion classifier was used for each genre. Each classifier was used to classify songs inside each genre. For example, if a song is classified as Jazz on the first layer, the emotion will be classified in the second layer using the emotion classifier trained with Jazz songs.

On the first layer, for genre classification, LIBSVM was used. To evaluate the multi-label classifier, macro average F-measure and micro average F-measure were used. Macro average treats each label equally and Micro average regards the overall prediction result across all labels.

The authors demonstrated that, even with the first layer genre classifier errors (58.98% accuracy), the accuracy on the second layer was higher than if the genres were known prior to the classification. The two-layer model had a macro average F-measure of 43% against 31% of the traditional model; and micro average F-measure of 56% against 52% of the traditional model.

**Automatic Classification of Musical Mood by Content-Based Analysis** (Laurier, 2011)

This thesis aimed to show the relationship between genre and emotion in the songs to improve emotion classification models.

The dataset consisted in 81749 tracks enriched with emotion labels from Last.fm database. By running dissimilarity tests in the emotion clusters, the authors decided to use a model of 4 emotion clusters: happy, sad, angry and calm/relaxed. The genre of these tracks was added through information sourced from the iTunes Music Store. Most of these tracks, 34.49% to be precise, belonged to the 'Rock' genre. Other genres included Reggae, Jazz, Electronic and Classical.

The machine learning algorithm chosen was SVM and the first conclusion reached was the observation of a high association between emotion and genre. Both positive and negative emotion categories are positively correlated with genres.

On the next phase the authors created three classifiers, one for audio features-only, one genre-only where the genre was extracted from the audio by using genre-classification methods, and one for the combination of the former two. The combination yielded the best accuracy which shows that combining genre with audio-features, even if it is predicted from low-level audio features, improves the emotion classification model.

As it happens with other papers in MER, the use of Last.fm database can be problematic in terms of bias of the emotions used as 'ground truth', because anyone can add labels to the songs on the online platform.

**Exploring Mood Metadata: Relationships with Genre, Artist and Usage Metadata** (Hu & Downie, 2007)

The main objective of this work is to demonstrate the relationships between emotion and artist, genre, and usage metadata. Usage metadata refers to the situations where the songs are adequate (e.g., Go to sleep, Driving).

The primary dataset for this study was AllMusic, composed of 179 emotion labels across 7134 albums and 8288 song, and the secondary datasets were extracted from epinions.com (for usage metadata) and Last.fm (for external corroboration).

A categorical model was used to classify the emotions in which there were 5 clusters each with 5 emotions.

In order to test the significance of the relationship between emotion and artist, genre and usage, the authors decided to use Fisher's Exact Test (FET).

After testing the AllMusic dataset, the same study was conducted using the Last.fm dataset, to see if the results corroborated. The authors discovered that the genre-emotion and artist-emotion relationship can be generalizable to other datasets while the usage-emotion is more dependable and vulnerable to the vocabulary present in the datasets.

The authors concluded that the genre-emotion and artist-emotion relationships were robust and show promise on its use on the MIREX task. On the other hand, the usage-emotion relationship is not robust enough to be used.

As reported on the paper, the AllMusic has a data sparseness problem since some emotions are represented on more than 100 albums while others are associated with only 3 albums or songs.

### 2.3.3. MER Datasets Review

In MER, there is not a consensus on which dataset is the best. Emotion annotations is labor intensive and time consuming (Chen, Yang, Wang, & Chen, 2015). This is one of the reasons why it is

difficult to compare different MER studies, since researchers do not use the same dataset. The most used dataset for benchmarking is the MIREX dataset but many researchers have raised questions about the taxonomy used. We will consider the datasets available in order to choose one for our analysis.

The evaluation criteria of the datasets will be focused on the following parameters:

- the taxonomy used to classify the data;
- if songs sampled and features used are public, if there is no access to the samples and features used but only to the conclusions or a mix in which the features are public but not the samples of the songs;
- if the emotion is annotated by segments of time or if it is continuous. MEVD are continuous real-time annotations usually using a window of one second or smaller while segments divide the song/sample in bigger time scales to perform the annotations;
- if the emotion annotations are created from perceived or induced emotions.

**MediaEval Database for Emotional Analysis in Music (DEAM):**

The largest public dataset for MER, DEAM is composed of 1802 songs (58 full-length songs and 1744 excerpts of 45 seconds) (Aljanaki, Yang, & Soleymani, 2016). The excerpts have a sampling frequency of 44100Hz and were sampled from a randomly, uniformly distributed, starting point in the songs. The annotations have a 2Hz sampling rate (Soleymani, Aljanaki, & Yang, 2016). The perceived emotion annotations were partially done in the lab and on the Amazon Mechanical Turk[7] crowdsourcing platform. The dataset contains music genres such as rock, country, jazz, electronic, etc.) and the data underwent thorough transformation and cleaning procedures. The dataset was used by a total of 21 teams from 2013 to 2015 on the task 'Emotion in Music' at the MediaEval Multimedia Evaluation Campaign. In the first two years of the task, the annotations were made in a window of 45 seconds, which made possible static and dynamic ratings. To capture more emotion variations, the 2015 dataset contained full-length songs. Since the dataset is public, both the songs and features are publicly available. The emotion taxonomy used is the numerical values representation of Russel's model of Valence-Arousal. In 2013 and 2014, each excerpt was annotated by at least 10 workers, while in 2015 each excerpt was annotated by a minimum of 5 workers. In 2015, these workers had to pass a filtering that excluded poor quality workers. This filter involved answering to multiple choice questions and free form questions. There were two types of annotations: dynamic and static. The dynamic annotations were made in a -10 to 10-point scale in which the annotator had to rate the music while it was being

[7] https://www.mturk.com/

played in its entirety. The static scale followed a 9-point scale on the 45 second excerpt. To understand the annotators consistency, the researchers employed Cronbach's α. Cronbach's α is a coefficient of internal consistency and is popular in psychometric tests (Cronbach, 1951). This test is sensible to the number of items. The larger the list of items it tests, the higher the α. The static annotations were far more consistent than the continuous ones. The static annotations passed the threshold of 0.7, which is considered an acceptable agreement between annotators (Aljanaki, 2016).

We consider that this dataset is the one that fits our research needs (at least to some extent) but it also shows multiple limitations.

To observe the limitations, we went a step forward to analyze the dataset's annotations and the difference between worker's annotations. First, since the static annotations were made in a 9-point scale for each Arousal and Valence, the standard deviation for each annotation was considerable. Arousal and Valence had 1.46 and 1.50 respectively. This fact led us to investigate further into the dataset to analyze what was the agreement between annotators and in which Quadrants, Genres and Artists resided the most disagreements. We started by calculating the Agreement Rate (%) for each song as shown in Equation 1:

$$\frac{No.\,of\ Annotators\ in\ favor\ of\ the\ Final\ Quadrant\ -\ No.\ of\ Annotators\ against\ the\ Final\ Quadrant}{Total\ Annotations\ for\ the\ Song}\ x\ 100$$

*Equation 1 – Annotations Agreement Rate (%)*

If the final annotated quadrant of a song is quadrant 4, and out of 5 annotators the 5 voted for the quadrant 4, the resulted Agreement Rate is 100%.

The dataset's mean of the Agreement Rate is 0.47 which shows that most songs had problems with disagreement between annotators. To understand better which songs were most troublesome, we filtered the dataset to only the songs that had less than 51% of Agreement Rate. The result were 1133 songs, which is 67.41% of the total dataset.

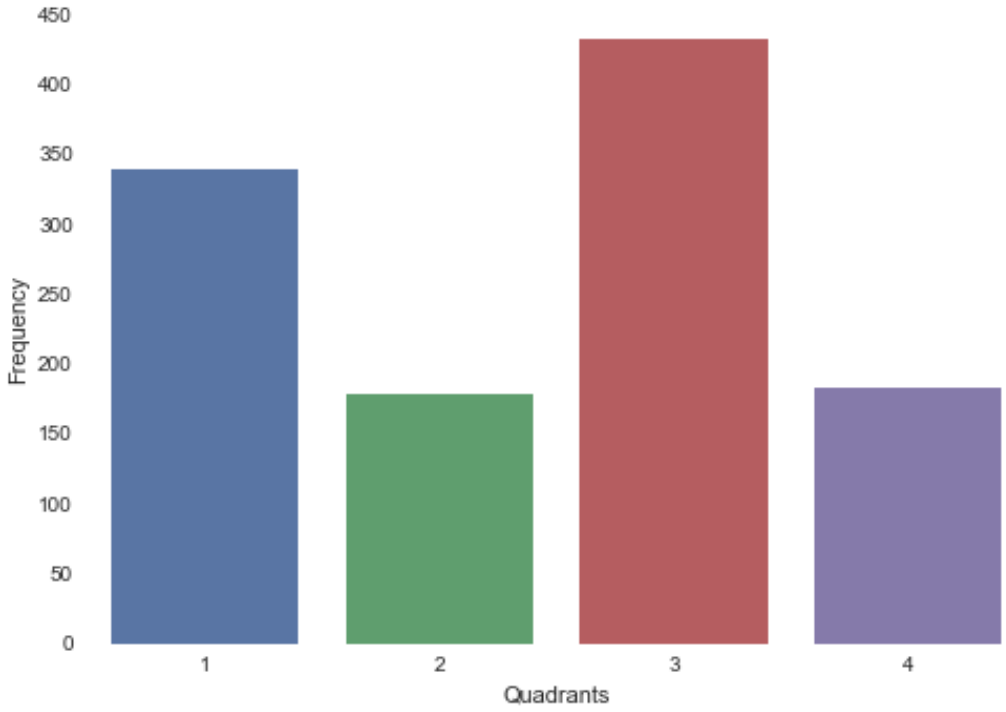



*Figure 4 - Quadrant distribution of songs with less than 51% Agreement Rate*

As we can see in Figure 4, most songs are represented in the First and Third quadrants. While Quadrant 1 and Quadrant 3 might appear to be the most represented, they decreased their representation when compared to the whole dataset. Quadrant 2 and Quadrant 4 had an increase in 4 percentage points in representation when compared to the whole dataset, from 12% to 16% in each Quadrant.

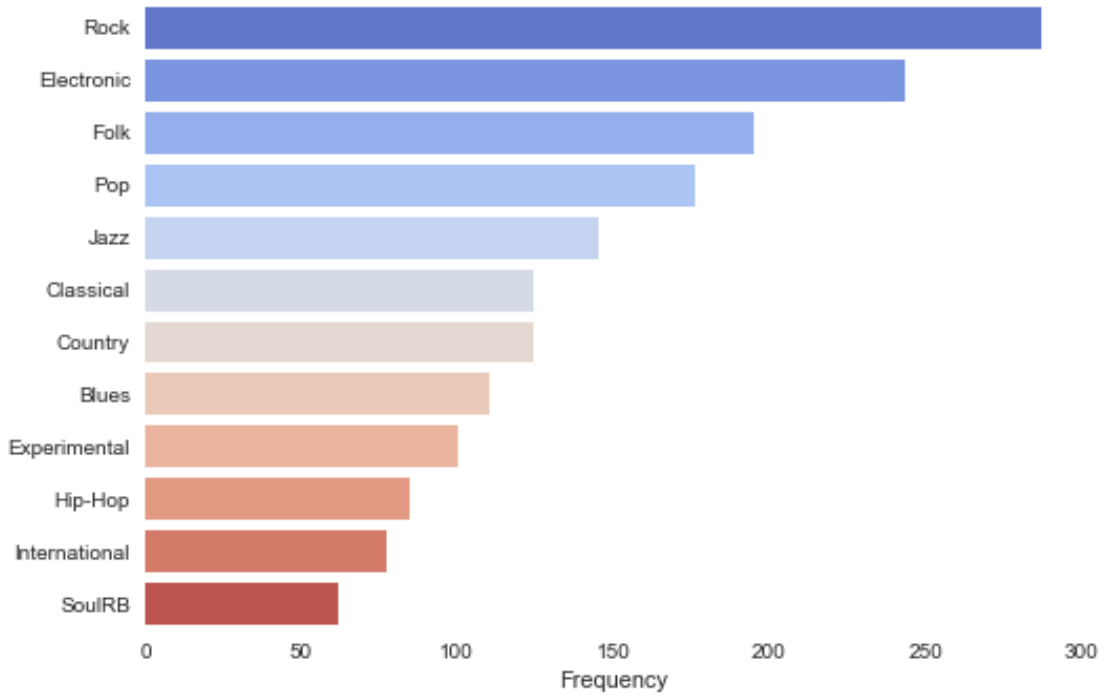



*Figure 5 - Genre distribution of songs with less than 51% Agreement Rate*

In Figure 5 we plotted the genre breakdown of the most 'troublesome' songs that were filtered before. When compared to the original dataset, the three genres that increased the most in representation were Hip-Hop (1,5%), Pop (0,79%) and Folk (0,67%).

To finalize the breakdown of the dataset, we analyzed if there were Artists which had a special incidence in these songs. It was interesting to note that there was an increase in representation of artists that only have one song in the entire dataset. Their representation increased in 5pp and the result is that 40% of the songs that generated more disagreement were composed by artists which had only one song in the dataset.

Secondly, static annotations, which are the focus of our research, were annotated on 45-second excerpts which were selected randomly. This can bring the problem of selecting the intro of a song that can be complete silence. Additionally, the scope of our research is to use the dominant emotion static annotations and the fact that the excerpt is 45 seconds, which is too long, can bring the problem of not having emotional stationarity, which means that one excerpt can have different dominant emotions that are placed in different quadrants, creating ambiguity in the annotation. We selected a small random sample of 20 songs and listened to them. Some songs in this sample had more than one dominant emotion as we suspected. Our annotations and the ones in the dataset were in agreement in 80% of the sampled songs.

Thirdly, the annotations were made using listeners from Amazon Mechanical Turk which can jeopardize the quality of the annotations, even when using a filtering phase for worker selection. To confirm the quality of the annotations, we listened to another random sample of 20 songs, now from the entire dataset instead of restricting the sample selection to the group of songs with lowest Agreement Rate, in order to compare our annotations with the ones made by the original workers. We only disagreed in 4 excerpts but it is important to note that some samples consisted mostly by applauses by the public, since the song was a live performance.

Furthermore, there is an imbalance of quadrant observations, genre observations and artist observations. Later, we will dive into the frequency of observations in genre and artist variables but for now we will focus on the quadrant imbalance. Quadrant 1 has 591 observations and Quadrant 3 has 681 observations. This contrasts with the 200 observations of Quadrant 2 and the 209 observations of Quadrant 4. This imbalance can lead to dubious results of the classifiers. We will dive into this in the implementation chapter.

Lastly, to check if our audio features, which we will dive deeper later, are in the state-of-art level we replicated the baseline model that was presented by the researchers (Aljanaki, 2016). Note that the original model used dynamic annotations while we use static annotations. The model consists of a linear regression using 20 x 10-fold cross-validation to predict Arousal and Valence values in Russel's continuous emotion plane. Both dependent values were scaled to [0,1]. The audio features used in the paper differ from the ones we used, which leads to different results. The features used by the researchers were a combination of Kalman filter and low level audio features: MFCC's, Zero-Crossing Rate, Spectral Flux, Roll off, Centroid and Spectral Crest Factors. The RMSE and Pearson's R were calculated for each fold and the average and standard deviation of inter-fold RMSE and Pearson's R were presented.

| | **Arousal** | | **Valence** | |
|---|---|---|---|---|
| | **RMSE** | **Pearson's R** | **RMSE** | **Pearson's R** |
| (Aljanaki, 2016) | .14 ± .06 | .37 ± .26 | .18 ± .09 | -.01 ± .39 |
| Replicated Model | .15± .0003 | .66 ± .001 | .14± .0003 | .64 ± .002 |

*Table 3 - Original versus Replicated Model results comparison*

On Table 3 it is possible to compare the results from both baseline models. In order to facilitate the interpretation, we will call the researchers model as original model and ours as replicated model. In the original model, both Valence and Arousal show a low average RMSE, which is a good indicator for model accuracy. In the replicated model, the RMSE for Arousal is similar and lower in Valence. The main strength of the replicated model when compared to the original one, is the Pearson's R which is

0.66 for Arousal and 0.64 for Valence, while the original model has 0.37 and 0.18 respectively. This means that the correlation between the predicted values and the original values in the replicated model is higher which shows a better capacity for classification of this dataset. Furthermore, the standard deviation values for both Arousal and Valence in the original model is high, showing symptoms that the classification model is volatile from fold to fold when compared with the replicated model.

Unfortunately, this was the only way to compare our model with other models which classified the whole DEAM dataset, but we have to keep in mind that static annotations usually show better results than dynamic annotations. With this analysis, we demonstrated that our audio features can keep with state-of-art research. As stated before, we will dive deeper into the detailed explanation of these audio features later.

**AllMusic:**

The AllMusic dataset consists in 1608 music clips each 30 second long. These songs were annotated by 665 subjects regarding valence and arousal. It is the second largest dataset in MIR. Researchers opted for a dimensional framework where the listeners were asked to annotate perceived emotions on the valence-arousal space. The western music song list was selected from AllMusic and the respective 30-second sample was extracted from 7digital. The songs were annotated by 665 listeners from Amazon Mechanical Turk. To ensure annotation quality, it was required that the subjects were residents of the United States and had over 90% tasks done on Amazon Mechanical Turk. To measure inter-subject agreement on the annotations, the researchers calculated Krippendorff's $\alpha$ which resulted in 0.31 on valence and 0.46 on arousal, which are considered fair agreement. (Chen et al., 2015). The main limitation of the dataset is the fact that it is partially public which makes impossible to extract new features from the songs. The annotations were made using listeners from Amazon Mechanical Turk which can, as stated before, reduce the quality of the annotations. Lastly, most part of the annotations fall in the first quadrant of the arousal-valence space, which means there is a quadrant imbalance on the dataset.

**Other Datasets:**

Besides the two datasets mentioned above, which are the largest ones in MIR research, there have been other datasets developed by researchers in order to conduct the studies.

The Soundtrack dataset, which consists of 110 film music excerpts of approximately 15 seconds each. The dataset is public and the annotations are based on perceived emotions of 116 non-musicians. In regards of the taxonomy, Eerola et al. (2011) used two different taxonomies to compare the results: half of the excerpts were representative of five discrete emotions (anger, fear, sadness, happiness and tenderness) and the other half were representative of three dimensions (valence, energy arousal and tension arousal). There was also released a set of 16 one-minute examples of emotions induced by music (Eerola & Vuoskoski, 2011). The main issues with this dataset is the small size and the fact that the excerpts are extracted from films which have considerable differences when compared to songs.

To conduct their research on semantic computing based of emotions on social tags, Saari & Eerola (2014) gathered a set of 600 randomly selected songs from Last.fm. The researchers used balanced random sampling to ensure that there was genre and emotion coverage. The sample also favored tracks associated to many emotion tags and with many listeners. There was also a restriction of having only one song per artist. Rock and Pop were the most frequent genres on the dataset. To annotate 15-second music clips of the songs, 59 participants were selected. Of these participants, 28 were musicians and 8 were trained professionals. The experiment was focused on emotions perceived and the listeners were asked to annotate the samples on bipolar terms: negative/positive, calm/energetic and relaxed/tense. To capture the continuous nature of emotion on songs, the results were converted to nine-step Linkert-scales (Saari & Eerola, 2014). One issue of the dataset is that the researchers disclosed the annotations but not the samples and features. This means that future researchers are not be able to replicate the results. Likewise, another main characteristic of the dataset, which makes unfeasible to our research, is the fact that the emotion annotations are continuous, while our focus is on the dominant emotion of the song. We could achieve the dominant emotion but it would create an additional layer of error to the annotation. Another limitation is the restriction to only on song per artist, which constraints any potential meaning of the variable Artist.

The Emotify dataset is composed of 400 one-minute song excerpts in four different genres (pop, classical, rock and electronic) collected through an online game named Emotify. The dataset is public and the annotations were made using GEMS scale (Geneva Emotional Music Scales), which is a taxonomy for induced emotions. Each annotator could skip the songs and switch musical genres because, as stated by the researchers, induced emotions don't occur in every song. The songs were split into two subsets, one which averaged 48 annotations and the other 16 annotations. What invalidates the application of this dataset on our research is the fact that it focuses on induced emotions while ours is on perceived emotions. Other issue is that, because the annotators could switch songs and

genres, the lesser known songs had fewer annotations when compared with popular music (Aljanaki, Wiering, & Veltkamp, 2016).

MoodSwings public dataset has 240 fifteen-second music clips which were selected to represent the four quadrants of the Arousal-Valence space. The annotations were made by participants online by listening to 30-second music clips and referring the dynamic position in the Arousal-Valence space. Later, the annotations were compared to other annotations developed using Mechanical Turk on the same music clips, which shown both sets of annotations were highly correlated (Schmidt & Kim, 2011). The small size is the main limitation of this dataset.

The Multi-modal MIREX-like emotion dataset was developed by Panda et al. (2013) by combining information from audio, lyrics and MIDI. The researchers used the MIREX framework to annotate 903 thirty-second audio clips, 764 lyrics and 193 MIDIs. The annotations were made resorting to the AllMusic database API (Panda, Malheiro, Rocha, Oliveira, & Paiva, 2013). This dataset is public but have some limitations. The first problem arises from the fact that it extracts the annotations from the AllMusic database, which have some quality problems since the annotation process is not clear (Malheiro, 2016). The only information is that these annotations are made by experts (Hu & Downie, 2007). Some of the excerpts were music intros or individuals talking. There is also the need to add an extra verification step when extracting labels from Last.fm. On their study, the verification step was made by 17 listeners, mainly students and researchers at the Music Technology Group (Laurier, 2011). On top of that, the MIREX framework has its own limitations, such as clusters overlapping semantically (Laurier et al., 2008).

### 2.3.4. MER Audio Frameworks Review

Audio feature extraction is an essential step in MER. Researchers commonly analyze the audio features present on the audio signals to reach meaningful conclusions on which emotion the song represents. There are audio frameworks which simplify the audio processing tasks and concede the opportunity of creating advanced applications based on them. We examine some of the most important ones. Most of these audio analysis tools were developed for other areas of MIR, and what differentiates these frameworks are the number and type of features available, performance, stability, system resources required and user-friendliness.

**Marsyas**

Marsyas or Music Analysis, Retrieval and Synthesis for Audio Signals, is an open-source software framework created by George Tzanetakis with the collaboration of students and researchers from around the world for audio processing with special application on the Music Information Retrieval field. One of the first frameworks in MIR, Marsyas is known for its computational efficiency, expected from a highly-optimized software written in C++. It has a solid user and developer base. Marsyas provides the structure to develop full GUI applications through the native integration with Qt14. It is important to note that the high number of features is due to the statistical measures calculated for each original feature. Some limitations are linked to the unsubstantial documentation, interface and syntax which require a steep learning curve, and difficult control of the audio processing networks.

**MIR Toolbox**

The MIR Toolbox is an integrated set of functions written in MATLAB for the extraction of musical features such as rhythm, pitch, timbre and others. Designed in a modular way, the framework allows the decomposition of different algorithms into more elementary functions. The main advantage is the possibility of combining these modules into new features by using novel approaches (Lartillot & Toiviainen, 2007).

The framework can be scripted or used with an interface and offers capability of extraction of a considerable number of relevant low and high-level features to the field of MER (Panda & Paiva, 2011). The creators gave special attention to the ease of use of the software and syntax. The documentation is pleasant when compared to other frameworks and there is the possibility of exporting and visualizing the extracted information. The main disadvantage is its reliance on MathWorks'MATLAB and MathWorks'Signal Processing Toolbox on account of being commercial products of this company. Contrary to Marsyas, MIR Toolbox does not offer real-time capabilities (Lartillot & Toiviainen, 2007).

**jMIR**

jMIR, an open-source package developed using Java, was created at McGill University to extract high and low-level features from audio files as well as to develop and share new features. The initial goal of the framework was to eliminate the effort in calculating from the signals by providing a broad range of algorithms suitable to MIR tasks. The application graphic user interface (GUI) is user-

friendly. It is possible to create user scripts for batch processing and other features through the command line interface. Other advantages include the higher number of audio features available. The fact that jMIR was built using Java makes it slower and heavier than other frameworks but also more portable. Regarding the feature extraction process, jMIR provides components such as jAudio, jSymbolic and jLyrics.

jAudio is the component of jMIR which gives the ability of extracting features from audio files. A relevant aspect of this tool is the possibility of combining low-level features to build high-level features.

jSymbolic is used to extract high-level musical features from symbolic music representations such as MIDI files. It is commonly used by researchers in empirical musicology, music theory and MIR, mostly for the high-level musical features related to instrumentation, rhythm, dynamics, melody, chords, texture and pitch statistics.

Jlyrics is a set of software tools for mining lyrics from the web and extracting features from them.

**LibRosa**

LibRosa is an open-source Python package created by Brian McFee for processing audio and music signals. It was designed in a way that does not pose a steep learning curve, especially for researchers familiar to MATLAB. Functions are designed to be modular, providing users with the possibility of creating their own custom functions. The integration with the Python package Matplotlib provides researchers with good visualizations of the rendered audio data (McFee et al., 2015). The ease of use and quality documentation are two other strengths of this software.

On the other hand, Librosa's output will only allow users unstructured data in table format while other softwares such as MIR Toolbox and Marsyas can also produce ARFF files to ease the data mining task.

Time lag is one of the weaknesses of Librosa, since Moffat et al. (2015) analyzed ten different audio feature extraction libraries and found that Librosa took 1h53min to process 16.5 hours of audio, while Marsyas completed it under 10min, JAudio under 15 minutes and MIR Toolbox took a little over 31 minutes (Moffat, Ronan, & Reiss, 2015).

### 2.3.5. Audio Features Review

#### 2.3.5.1. Intensity

**Root-Mean-Square Energy (RMS):**

Root-Mean-Square Energy measures the power of a signal over a window of time. It can also be used to understand the global energy of a signal $x$ by taking the root average of the square of the amplitude (RMS) (McEnnis, McKay, & Fujinaga, 2005), as shown below (Equation 2):

$$x_{RMS} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} {x_i}^2} = \sqrt{\frac{{x_1}^2 + {x_2}^2 + \cdots + {x_n}^2}{n}}$$

*Equation 2 – Root-Mean-Square Energy*

**Root-Mean-Square Derivative:**

Indicates the change of signal power by representing the window-to-window change in RMS.

**Root-Mean-Square Variability:**

Gives the value of the standard deviation of the RMS on the last N windows.

**Less-Than-Average Energy**

In order to understand if the energy distribution remains constant throughout the signal, researchers can use the percentage of frames showing less-than-average energy, also known as low energy rate (Tzanetakis, 2002), as shown in Figure 6.

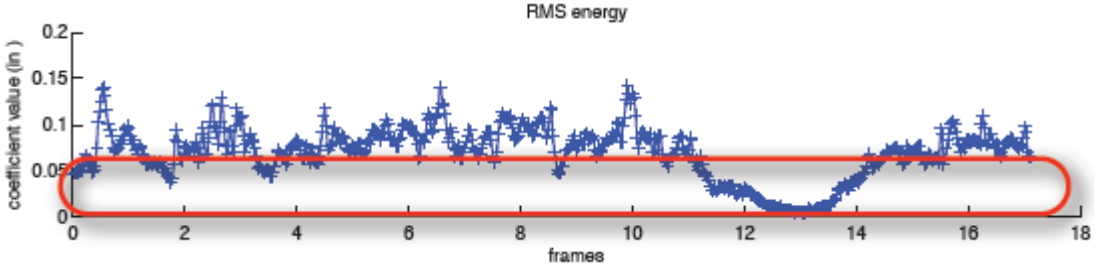



*Figure 6 - Lower-than-average energy frames highlighted on Energy curve* (Lartillot, 2013)

**Fraction of Low Energy Frames:**

As an indication of the variability of windows, researchers define the fraction of the 100

previous windows in which the RMS is below the mean RMS, thus finding which are the windows whose signal is quiet relative to the rest of the signal section.

#### 2.3.5.2. Rhythm

**Rhythmic Fluctuation:**

Indicates the rhythmic periodicity along auditory channels. The process of the estimation of these feature is based on two steps (Lartillot, 2009):

- First the spectrogram is computed on 23ms frames and half overlapping, then the Terhardt outer ear is modelled with Bark-band redistribution of energy and estimation of the masking effects. The amplitudes are then computed in a dB scale.

- After that, the FFT, which varies from 0 to 10 Hz, is computed for each Bark band. The final modulation of coefficients is based on a psychoacoustic model of the strength of the fluctuations. This results in a matrix filled with rhythmic periodicities of each Bark band.

**Tempo:**

Indicated by beats per minutes (BPM), measures the pace of a music piece and it is usually calculated by identifying the periodicities from the onset detection curve.

**Strength of Strongest Beat:**

Uses the beat histogram to identify the strongest beat and compare it to other beats (McKay, 2005).

**Beat Sum:**

Measured by the sum of all bins in the beat histogram, it is commonly used to assess the importance of regular beats on a given signal (McKay, 2005).

#### 2.3.5.3. Timbre

**Attack Time:**

Estimation of the duration for a signal to reach to its peak, as shown in Figure 7.

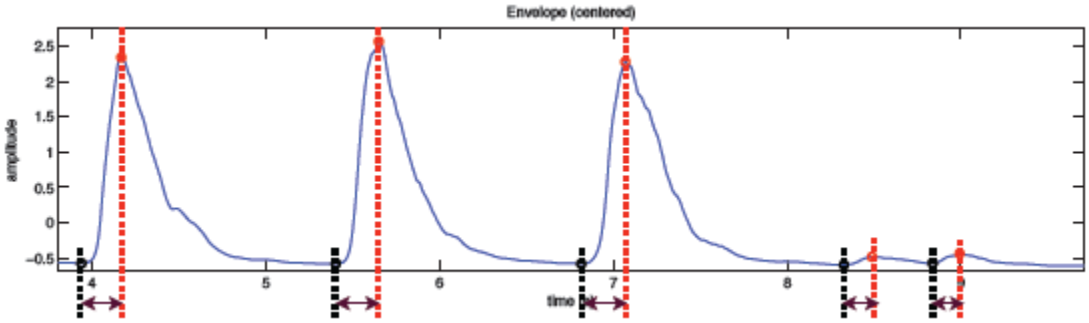



*Figure 7 - Attack Time detection* (Lartillot, 2013)

**Attack Slope:**

Other measure used to indicate the attack phase, where the average slope of the whole attack phase is calculated since it starts until it reached the peak as shown in Figure 8.

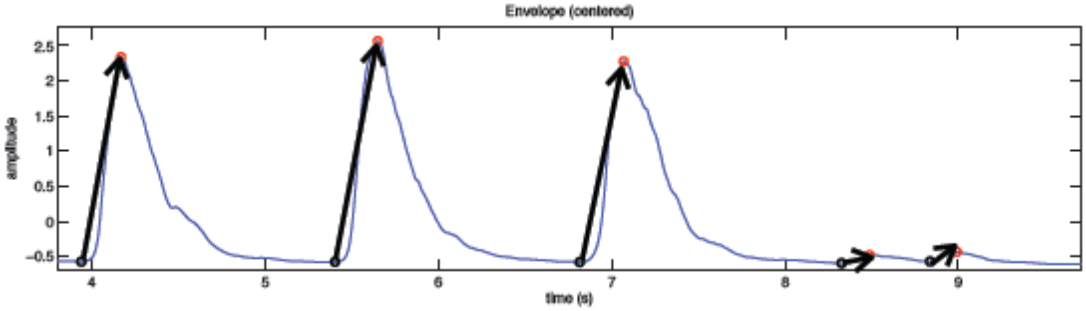



*Figure 8 - Attack Slope Example* (Lartillot, 2013)

**Zero Crossing Rate:**

Consists on the count of times the waveform changes sign by crossing the horizontal axis, as shown on Figure 9. Commonly used to indicate the noisiness of a song.

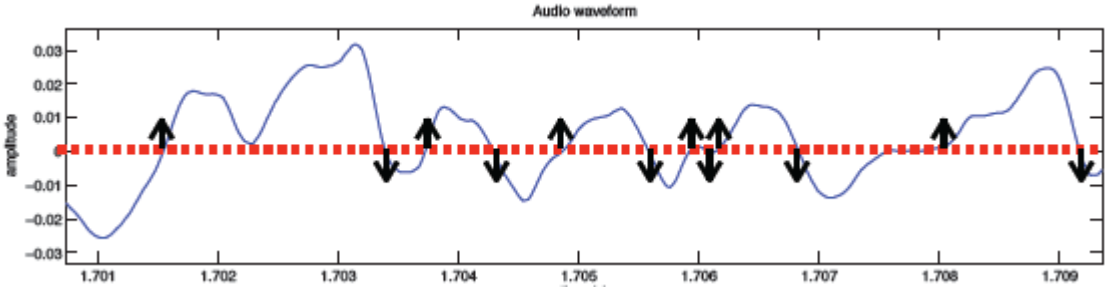



*Figure 9 - Zero Crossing Rate Waveform* (Lartillot, 2013)

As shown below (Equation 3), Zero Crossing Rate is calculated using a formula in which $\boldsymbol{sign}$ is a function where 1 is for positive arguments and 0 for negative arguments. $\boldsymbol{x[n]}$ is the time domain signal for frame $\boldsymbol{t}$ (Tzanetakis, 2002).

$$Z_t = \frac{1}{2}\sum_{n=1}^{N} |sign(x[n]) - sign(x\,[n-1]|$$

*Equation 3 – Zero Crossing Rate*

**Zero Crossing Derivative:**

Estimation of the absolute value of the change from window-to-window in Zero Crossing, indicating noisiness and frequency.

**Spectral Roll Off:**

Commonly used to indicate the skew of the frequencies of a window, it consists on the fraction of the total energy in Hz that is below a given percentage threshold, as exemplified in Figure 10. Usually this threshold is set to 85% (Tzanetakis, 2002).

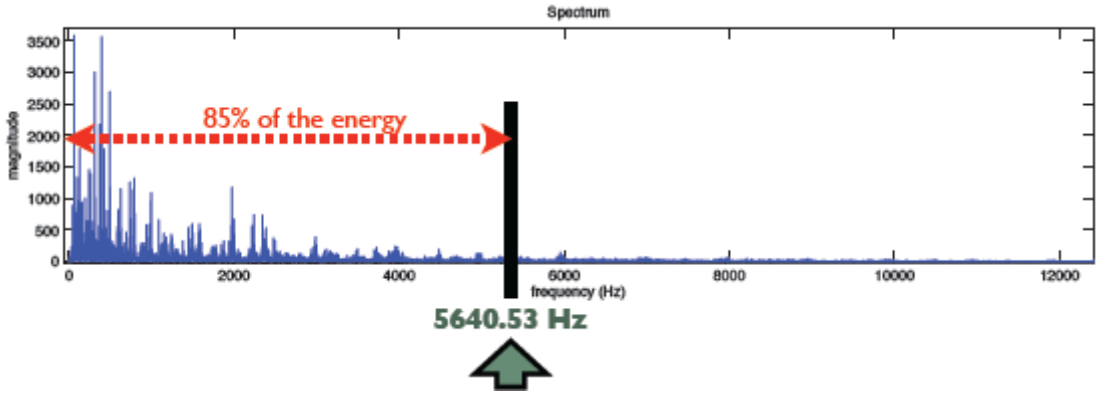



*Figure 10 - Spectral Roll Off with frequency* (Lartillot, 2013)

The Spectral Roll-Off, according to (Tzanetakis, 2002), is shown on Equation 4. $\boldsymbol{R_t}$ stands for the frequency below which 85% of the magnitude distribution is condensed.

$$\sum_{n=1}^{R_t} M_t[n] = 0.85 * \sum_{n=1}^{N} M_t[n]$$

*Equation 4 – Spectral Roll Off*

**High Frequency Energy:**

Consists in fixing a minimum frequency threshold and measuring the amount of energy above that value, as shown in Figure 11. The thresholds for the frequency is usually 1500HZ (Lartillot, 2009). The result is expressed between 0 and 1.

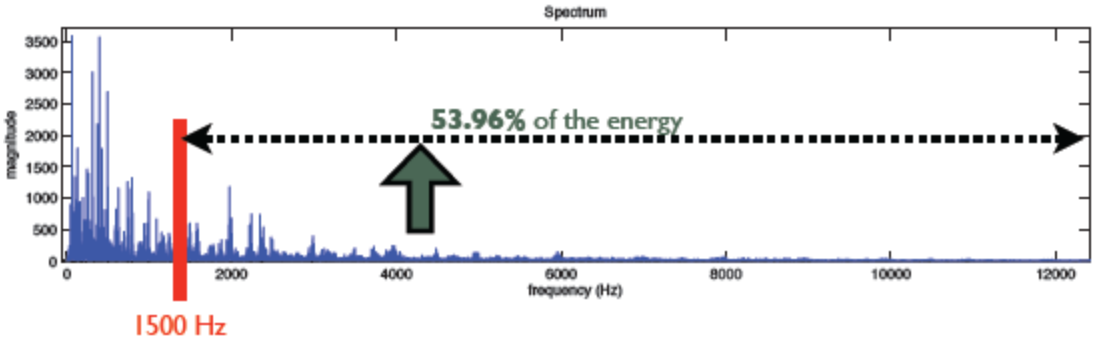



*Figure 11 - Spectral Roll Off using Percentage* (Lartillot, 2013)

**Spectral Flux:**

Measures the distance between adjacent frames. Musical experiments showed that it is an important attribute for the listener's perception of musical instrument timbre (Tzanetakis, 2002).

Spectral Flux can be calculated as shown in Equation 5, where $\boldsymbol{N_t[n]}$, $\boldsymbol{N_{t-1}[n]}$ are the normalized magnitude of the Fourier transform at the current frame $\boldsymbol{t}$, and the previous frame $\boldsymbol{t-1}$ (Tzanetakis, 2002).

$$F_t = \sum_{n=1}^{N} (N_t[n] - N_{t-1}[n])^2$$

*Equation 5 – Spectral Flux*

**Mel-Frequency Cepstral Coefficients (MFCC):**

Describe the spectral change of sound. The frequency bands are positioned on the Mel scale logarithmically. This approximates the human auditory response which is closer to reality than the linearly-spaced frequency bands.

To calculate MFCC, first we need to take the log-amplitude of the magnitude spectrum. Next, the FTT bins are grouped and smoother according to the Mel-scale. To eliminate the correlation between the resulting feature vectors, a Discrete Cosine Transform is used (Tzanetakis, 2002).

**Sensory Dissonance (Roughness):**

As shown in Figure 12, Sensory Dissonance depends on the frequency ratio of each pair of sinusoids (Lartillot, 2009).

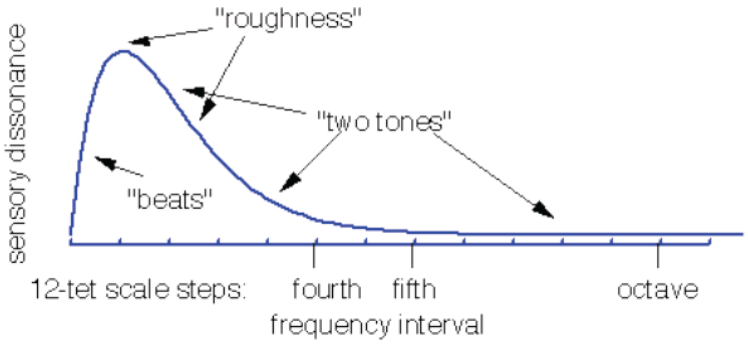



*Figure 12 - Sensory Dissonance depending on frequency ratio* (Lartillot, 2013)

**Spectral Peaks Variability (Irregularity)**

The degree of variation of the successive peaks of the spectrum. It can be calculated as shown in Equation 6, with the sum of the amplitude minus the mean of the preceding, current and next amplitude (Lartillot, 2009).

$$\sum_{k=2}^{N-1} |a_k - \frac{a_{k-1} + a_k + a_{k+1}}{3}|$$

*Equation 6 – Spectral Peaks Variablity*

**Spectral Centroid:**

Measuring the central shape, this feature indicates the "brightness" of the textures of the song. It can be calculated as following:

$$C_t = \frac{\sum_{n=1}^{N} M_t[n] * n}{\sum_{n=1}^{N} M_t[n]}$$

*Equation 7 – Spectral Centroid*

In the Equation 7 , $\boldsymbol{M_t[n]}$ represents the magnitude of the Fournier transform at frame $\boldsymbol{t}$ and frequency bin $\boldsymbol{n}$ (Tzanetakis, 2002).

**Linear Prediction Reflection Coefficients:**

Commonly used in speech research as an estimate of the speech vocal tract filter but also used in musical signals (Tzanetakis, 2002).

**Linear Spectral Pairs:**

Represents linear prediction coefficients for transmission over a channel. This feature is not sensitive to quantization noise and it is stable, making them popular for speech coding.

**Strongest Frequency via Spectral Centroid:**

This feature is an estimation of the strongest frequency component of a signal (Hz), by using the spectral centroid (McKay, 2005).

**Spectral Crest Factor:**

Known also as peak-to-average ratio, this feature is a measure of a waveform, calculated from the peak amplitude divided by the RMS value of the given waveform as shown in the following formula (Equation 8):

$$C = \frac{|x|_{peak}}{x_{rms}}$$

*Equation 8 –Spectral Crest Factor*

**Spectral Flatness Measure:**

Characterizes an audio spectrum: if the graph of the spectrum appears flat and smooth, the Spectral Flatness Measure (SFM) is probably high. This happens because a high SFM means that a spectrum has a similar amount of energy across all spectral bands, while a low SFM describes a spectrum where the power is condensed in a small number of bands.

It can be calculated by dividing the mean of the geometric power spectrum by the arithmetic mean of the power spectrum (Equation 9). In the equation, $\boldsymbol{x(n)}$ represents the magnitude of bin number $\boldsymbol{n}$:

$$SFM = \frac{\sqrt[N]{\prod_{n=0}^{N-1} x(n)}}{\frac{\sum_{n=0}^{N=1} x(n)}{N}}$$

*Equation 9 –Spectral Flatness Measure*

**Inharmonicity:**

Inharmonicity measures the number of partials that are not multiples of a given fundamental frequency, as shown in Figure 13:

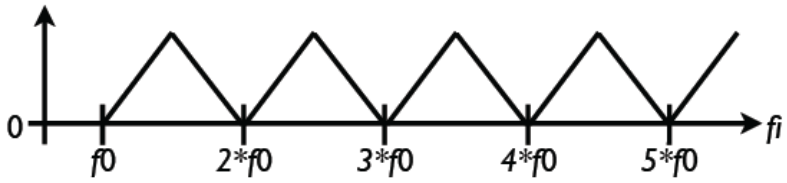



*Figure 13 - Fundamental frequency (f0) and respective multiples* (Lartillot, 2013)

### 2.3.5.4. Pitch

**Pitch:**

Indicates the perceived fundamental frequency of a given sound. Along with loudness and timbre, it is one of the three major auditory sound attributes. It represents the fundamental frequency of a monophonic sound signal (Tzanetakis, 2002).

**Strongest Frequency via FFT maximum:**

Estimation of the strongest frequency component in a given signal (Hz) found via FFT bin with the highest power (McKay, 2005).

### 2.3.5.5. Tonality

**Key:**

Estimation of tonal centre positions and their respective clarity (McKay, 2005).

**Mode:**

Indicates the difference between major and minor keys of a given piece.

**Tonal Centroid:**

Projects a 6-dimensional feature vector which corresponds to chords along circles of fifths and a major of thirds. By using the Euclidean distance on the Harmonic Network of Tonnetz, which represents the pitch relations, this feature finds the distance between successive analysis frames of tonal centroid vectors, detecting harmonic changes (Lee, 2008), as shown in Figure 14.

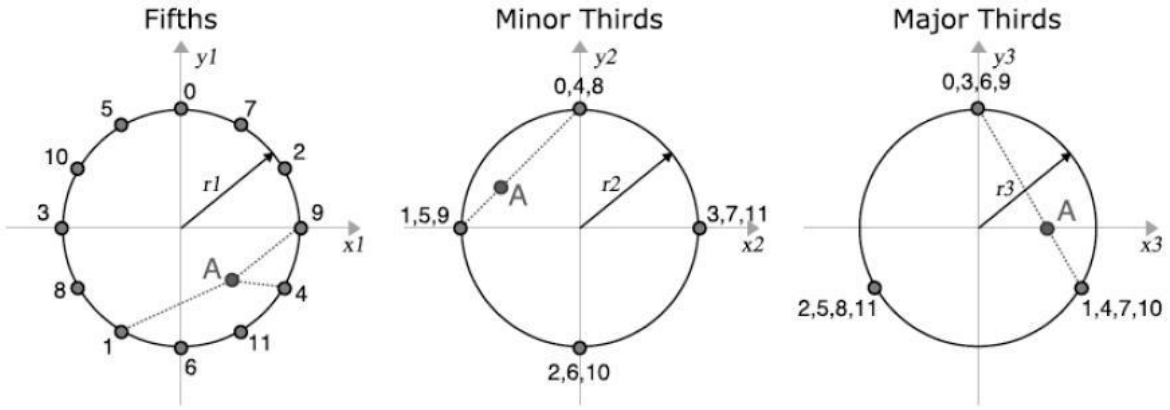



*Figure 14 - Tonal Centroid for a major triad is shown at point A (Lee, 2008)*

**Harmonic Change Detection Function:**

Represents the flux of the tonal centroid (Lartillot, 2009).

### 2.3.5.6. Musical Features

Divided in two distinct categories, rhythmic and pitch content, this features are based on musical content and information extracted from previous mentioned features (Tzanetakis, 2002).

**Rhythmic Features – calculated via Beat Histograms (BH) of a song:**

- A0, A1: relative amplitude of the first (A0) and second (A1) histogram peak;
- RA: ratio of the amplitude of the second peak and the amplitude of the first peak;
- P1, P2: period of the first (P1) and second (P2) peaks in Beats per Minute;
- SUM: sum of the histogram, refers to the beat strength.

**Pitch Features – calculated via Folded Pitch (FPH) and Unfolded Pitch (UPH) Histograms of a song:**

- FA0: amplitude of the maximum peak of the FPH, indicates the most dominant pitch class of a song;
- UP0: period of the maximum peak of the UPH, it refers to the octave range of the dominant musical pitch of a given song;
- FP0: period of the maximum peak of FPH. Corresponds to the main pitch class of a song;

- IPO1: pitch interval between the two most prominent peaks of the FPH, referring to the main tonal interval relation;
- SUM: sum of the histogram, indicates the strength of the pitch detection.

# 3. IMPLEMENTATION

## 3.1. Pre-processing the Dataset

We will use the DEAM dataset, which is composed of 1802 songs (1744 samples of 45s each and 58 full-length songs). As mentioned previously, the excerpts have a sampling frequency of 44100Hz and were annotated according to the continuous emotion taxonomy representation of Russel's model of Valence-Arousal.

From the original dataset, 9 songs were removed because there was no annotation associated with this samples. This represents 0.49% of the DEAM dataset.

We transformed the numerical values of the Russel's model of Valence-Arousal into four quadrants, which is the Categorical version of this emotion taxonomy (Malheiro, Oliveira, Gomes, & Paiva, 2016). We will dive into the process later. Since our dependent variable, Quadrants, is not numerical after the before mentioned transformation, this is considered a Classification Problem.

We used Python to implement most of the project, from Data Pre-processing until Data Visualization. There were other tools used that will be mentioned when necessary.

Our variables can be divided into 4 groups: quadrants annotated, audio features, genre and artist. Quadrants annotated is the dependent variable while the other groups consist of independent variables. We will dive into each one of these groups and explain which transformations were done. In the end, we will present the final dataset used to train the machine learning algorithms.

### 3.1.1. Quadrants

Our dependent variable, *Quadrants*, consists of static annotations on the dataset songs. The annotations were performed in a 9-point scale for Valence and Arousal. The emotion taxonomy followed by the researchers was a numerical representation of Russel's model of Valence-Arousal. The minimum was 1 and the maximum was 9 for Valence and Arousal and the final annotations for each song was the average of all the annotators rating. Each song was annotated by a minimum of 5 listeners.

We decided to transform this variable by placing each song into one of the four quadrants of Russel's model of Valence-Arousal. This is the Categorical Model approach of this emotion taxonomy. The meaning of each quadrant is the following (Malheiro et al., 2016), as shown in Figure 15: 1$^{st}$ Quadrant - Happy; 2$^{nd}$ Quadrant - Tense; 3$^{rd}$ Quadrant - Melancholy, 4$^{th}$ Quadrant - Serene Joy.

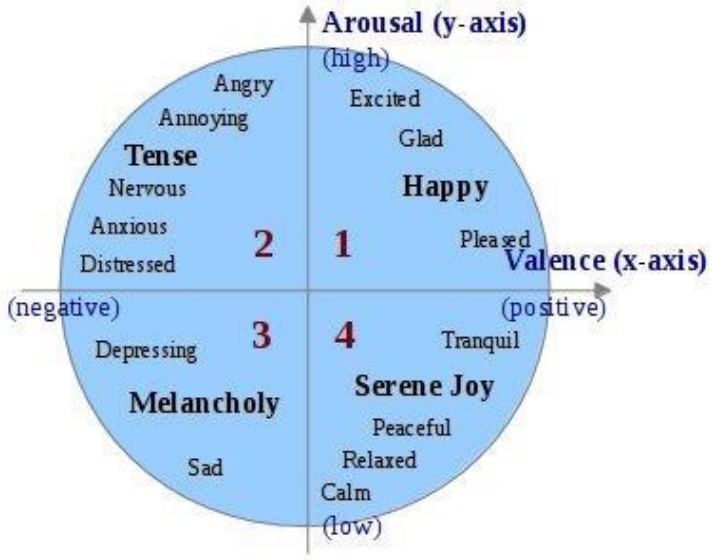



*Figure 15 - Russel's Circumplex Model (Y.-H. Yang et al., 2008)*

We did not round the numbers either up or down. To avoid ambiguity, the 108 songs that were placed on top of one of the axis were removed, which represents 6.02% of the total dataset.

When we analyze the songs distribution for each quadrant (Figure 16), it is easily identified a considerable class imbalance. Quadrant 1 and 3 combined represent approximately 76% of the total songs. This is far beyond the desirable 25% representation of each of the four quadrants.

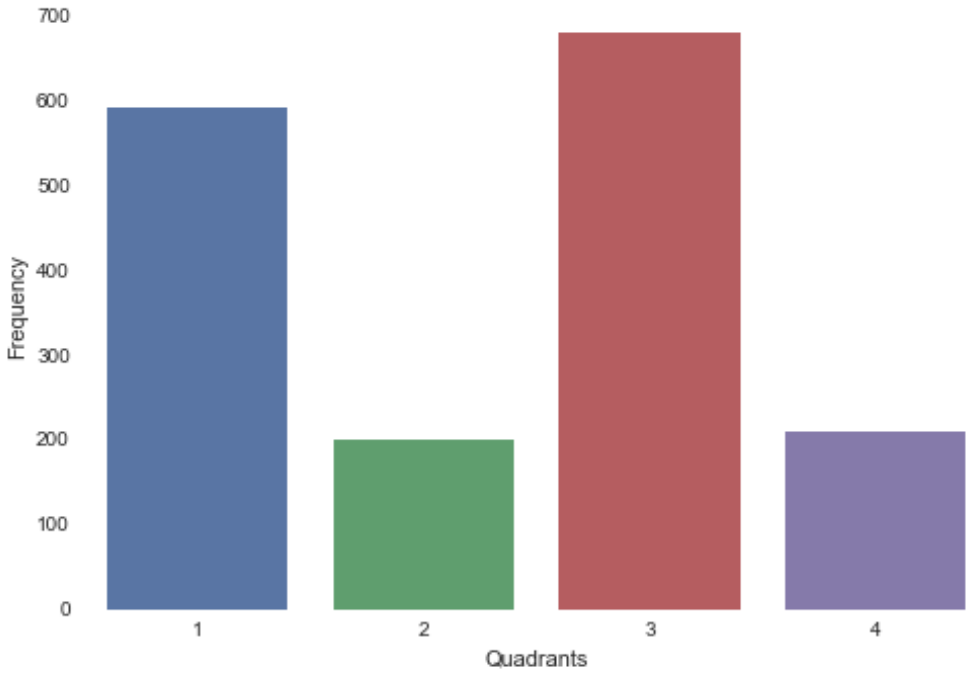



*Figure 16 - Quadrant distribution of songs*

The main issue with class imbalance in classification problems is that during training, the model ends up having a strong bias towards predicting the classes with more observations and will rarely predict the class with less observations. The stronger the imbalance, the more biased the model becomes. Because the model will pick classes with the most observations, the probability of getting the prediction right is higher, therefore the accuracy measure will appear to be higher too. One way to deal with this problem is to resample the dataset. Resampling balances the training dataset by increasing the percentage of representation of the minority classes and, consequently, decreasing the representation of the former majority classes. There are two major ways of resampling a training set: oversampling and undersampling methods. Oversampling methods involves the creation of artificial data records belonging to the minority classes, thus increasing their representation. On the contrary, undersampling balances the training set by removing data points which belongs to the majority classes, eliminating the gap between classes (A. Liu, Ghosh, & Martin, 2007).

Since we did not want to add any artificial data to our training set, because an important step in our research is to listen to the songs and understand why the model misclassified them, we opted for undersampling. It is important to note that the main disadvantage of undersampling is that it eliminates potential useful information (A. Liu et al., 2007).

The records that were removed were selected randomly from the majority classes, but the representation of the Genre and Artist remained consistent with the original model. The resampled dataset has a representation of 200 songs for each of the four quadrants, which was the number of songs that were present on the lowest frequency quadrant, Quadrant 2, of the original dataset.

### 3.1.2. Audio Features

The audio features used are the default features of the audio feature extraction software Marsyas (Panda et al., 2013). Each variable is divided into six 1st and 2nd statistical features: mean, standard deviation, kurtosis, skewness, min and max. The default extracted features are *ZeroCrossings, Centroid, Rolloff, Flux, MFCC0, MFCC1, MFCC2, MFCC3, MFCC4, MFCC5, MFCC6, MFCC7, MFCC8, MFCC9, MFCC10, MFCC11, MFCC12, PeakRatioChromaA, PeakRatioChromaA#, PeakRatioChromaB, PeakRatioChromaC, PeakRatioChromaC#, PeakRatioChromaD, PeakRatioChromaD#, PeakRatioChromaE, PeakRatioChromaF, PeakRatioChromaF#, PeakRatioChromaG, PeakRatioChromaG#, PeakRatioAverageChromaA, PeakRatioMinimumChromaA*.

The first transformation was to remove variables which had the same value for all the observations, making them redundant to the model.

Secondly, we ran a Person's correlation coefficient matrix in order to remove highly correlated variables. Collinearity refers to the non-independence of predictor variables, therefore increasing the perceived variance of the parameters which will mislead the researcher when identifying the most relevant predictors. Pearson's correlation coefficient measures the strength of the linear relationship between two variables, as shown below (Equation 10). There is an underlying assumption that the variables are continuous, which is the case for the audio features used (Hauke & Kossowski, 2011).

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{(n\sum x^2 - (\sum x)^2)(n\sum y^2 - (\sum y)^2)}}$$

*Equation 10 – Pearson's correlation coefficient*

As stated on (Mela & Kopalle, 2002), there is no consensus on the correlation threshold to better prevent collinearity. We removed the variables that had a correlation of more than or equal to |0.5| to eliminate collinearity and reduce dimensionality to 50 independent audio variables.

Thirdly, the outliers were removed. The removal of outliers is a common preliminary step of data analysis because most machine learning algorithms are sensitive to this data, therefore impacting the

results and respective assumptions. We followed a conservative approach, removing observations which were 5 standard deviations apart from the mean. The general assumption is that outliers have a low probability of being generated by the overall distribution. For example, in a normal distribution, approximately 68%, 95% and 99.7% of the data are within 1, 2 and 3 standard deviations from the mean, respectively (Seo, 2006). The decision of removing only data points that were at least 5 standard deviations from the mean was due to the fact that we have a small dataset and cannot afford to remove more data points and because the samples are 45 seconds-long, which is longer than we would like to, therefore it can have sudden peaks or changes of rhythm which would place most songs has outliers in at least one of the audio features.

Lastly, we normalized the audio variables. This step is crucial in Pre-processing because we need that the machine learning algorithms, for example K-Nearest Neighbours, weight each variable equally. If the scales of this variables vary wildly, as they do, the ones with wider scales will have an added weight on the model when compared with the ones with narrow scales. We decided to used Min-Max Normalization, which will transform every scale to an interval between 0 and 1, corresponding to the min and max of each variable respectively (Equation 11) (Patro & Sahu, 2015).

$$Min - Max\ Normalization = \frac{x - Min(x)}{Max(x) - Min(x)}$$

*Equation 11 – Min-Max Normalization*

### 3.1.3. Genre

Genre and Artist are two of the most important variables in our research, since one of the main goals is to study their impact on the song's Quadrant. We used the Genre that was present on the DEAM dataset, which was extracted from freemusicarchive.org and jamendo.com. These websites choose the genre that fits best to each song, according to their experts' opinion. We only kept genres that appeared in more than 1% of the songs, repeating the process of previous researches regarding Genre recognition (Laurier, 2011). In result of this process, 4 songs were removed because they had exclusively genres that were eliminated. Each song can have more than one Genre, which is the main reason why we created dummy variables for each genre. The Genres present on the dataset after the removal are: Rock, Pop, Soul/R&B, Blues, Electronic, Classical, Hip-Hop, International, Experimental, Folk, Jazz, Country and Instrumental. In order to avoid perfect multicollinearity, which occurs when more than two predictors are inter-correlated and one of the variables can be predicted from the

combination of the others, we removed Instrumental which was the Genre with less frequency. The main problem of perfect multicollinearity is that it invalidates the inferences that we can extract from the effects of collinear variables (Dormann et al., 2013).

To test correlation between the Genre dummy variables, we used Pearson's R and observed that there was no considerable correlation between the variables. The highest correlated were the genres Blues and Soul/R&B with a correlation coefficient of 0.41.

As shown on Figure 17, the most represented Genres in the dataset are Rock (17.65%), Electronic (14.32%) and Folk (10.91%). On the other hand, Soul/R&B (3.50%), International (4.18%) and Hip-Hop (4.77%) are the least represented.

In order to have an idea of what will be the impact of the Genre variables on the dependent variable, *Quadrants*, we analyzed the Pearson's correlation coefficient between the former and the genres. There were no correlation coefficients above 0.17, which was the Pearson's r value for the variables *Folk* and *Quadrants*.

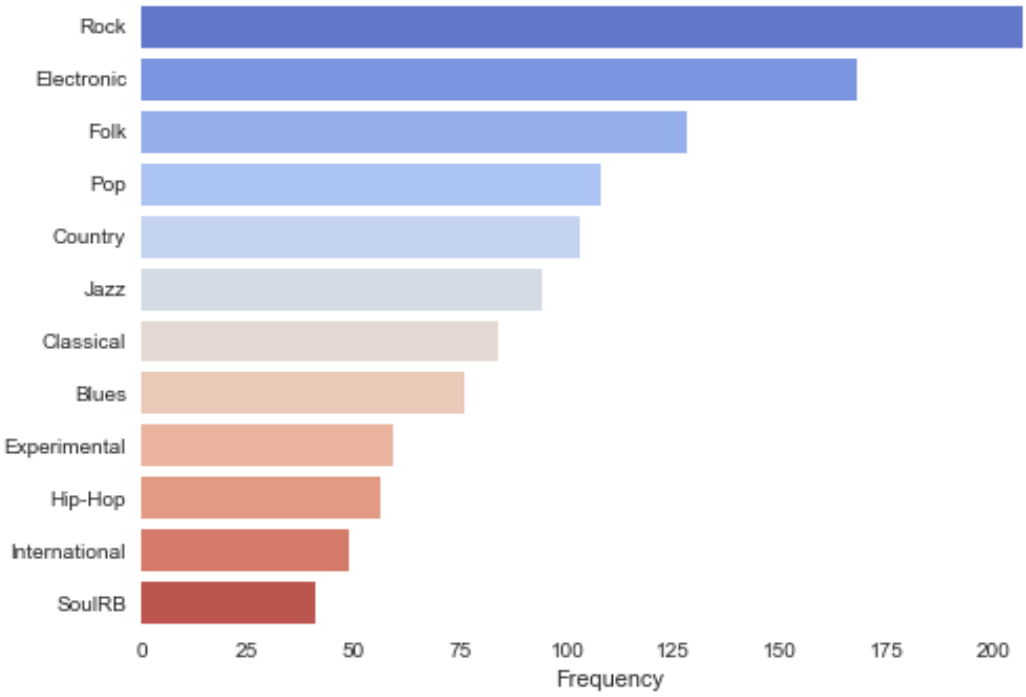



*Figure 17 - Genre distribution of Songs*

### 3.1.4. Artist

The Artist that interpreted each song was extracted from the original DEAM dataset, similar to the Genre. We approach the Artist variable in two ways:

- 1st Approach: In the case that an Artist has a song with another Artist, this combination will be considered as a new Artist on our model;
- 2nd Approach: In the case that an Artist has a song with another Artist, the song will be attributed to the Artist which owns the song, e.g. the song is in the Artist's album. This will lead to an increase of frequency of some Artists.

We will compare these two approaches by checking which model has the better performance and calculate if the difference is statistically significant. This will help us understand what is the best way to approach the Artist variable.

To understand the difference between both approaches, we will briefly analyze how many artists have repeated songs in the dataset before and after the transformation.

In total, after the Pre-processing phase, there are 525 artists represented in the dataset, when we consider the variable as stated in the 1st Approach. When we switch to the 2nd Approach, the number of artists decreases to 522, which shows that, in this dataset, the weight that the songs with featuring Artists have is not considerable. Nonetheless, we will analyse the impact since it can be useful for future researches where the dataset can have more songs with featuring artists.

As shown in Figure 18, 74.86% of the total artists have just composed one song in the dataset, 13.29% appeared in two songs and 5.52% composed three songs. The Artist that is more represented, Jason Shaw, has a total of 13 songs.

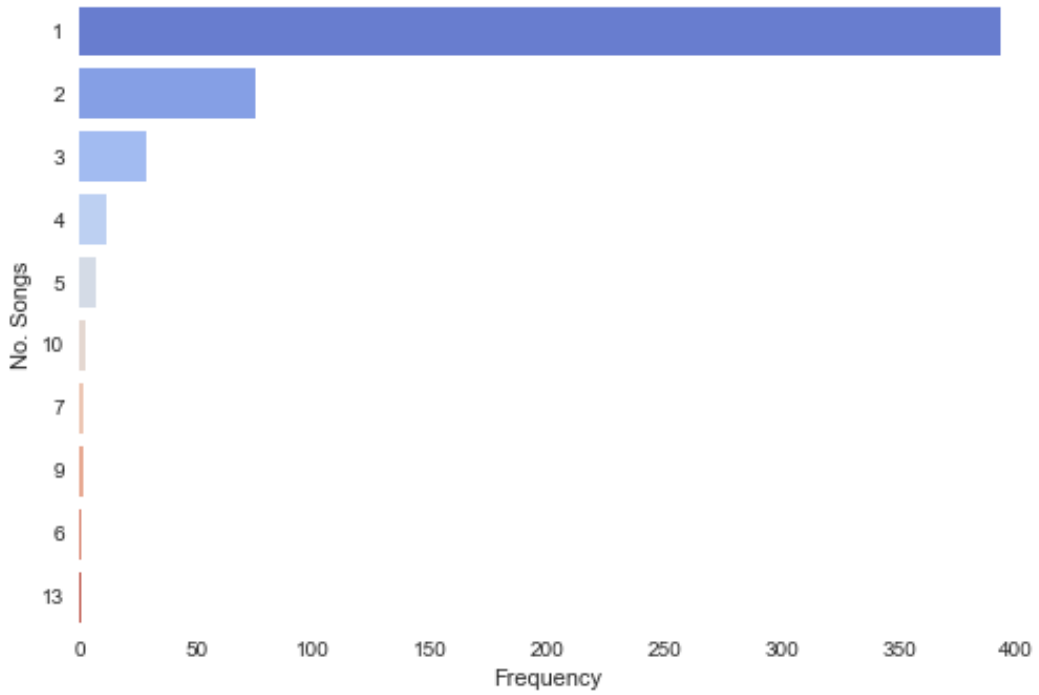



*Figure 18 - Number of Songs per Artist using 1st Approach*

When the variable followed the 2nd Approach, the total of Artists that only composed one song were 73.75%, which is a decrease of 1.1 percentage points when compared with the 1st Approach. The representation of Artists that composed 2 and 3 songs increased in 1.46 and 0.42 percentage points respectively. This can be explained by the fact that some combination of artists was being regarded as a new artist on the 1st Approach and on the 2nd Approach the song is being attributed to the main artist which had already other songs in the dataset. The maximum of songs recorded by one artist remained unaltered.

Since we will start our model with the 1st Approach to the Artist variable and only later will test the 2nd Approach, the 1st Approach will be the default and the other method will only be used when explicitly stated.

To facilitate the task of running the machine learning algorithms and other Pre-processing activities, we transformed this categorical variable into a numerical one. Instead of having the name of the artist, the variable will hold a number that works as an Artist ID which can later be tracked back into the name of the Artist.

To understand the impact between Artist and Quadrants, we calculated the Pearson's correlation coefficient between these two variables, which resulted in -0.04. This means that the variable Artist doesn't have a considerable impact on the resulting Quadrant by itself.

### 3.1.5. Correlation between Independent Variables

To understand if we are running into collinearity problems between Audio Features, Genre and Artist, we check if there was high correlation between these variables.

We started by using Pearson's correlation coefficient to understand what are the correlation values between Audio Variables, Dummy Genre Variables and Artist. There were no correlations above |0.5| which was our threshold for the audio features in the Pre-processing step done earlier in this research.

### 3.1.6. Principal Components Analysis on Audio Features

Since we have 50 audio variables, we tried to reduce dimensionality by running a Principal Components Analysis. The curse of dimensionality is a term called to the increased difficulty to classify, analyse and visualize high-dimensional data. This happens because the data points become sparser as the dimensionality increases. This can impact clustering techniques since it relies on distance and intra-cluster similarity (Steinbach, Ertöz, & Kumar, 2003).

Principal Components Analysis is one of the most popular techniques for dimensionality reduction and attempts to find a linear basis of reduced dimensionality on the data, where the variance of the former is maximized. Usually, the threshold of variance explained is 0.9 or 0.95 in order to accept the results of the technique (Van Der Maaten, Postma, & Van Den Herik, 2009). The results were not accepted, since even the three Principal Components that had the highest explained variance, accounted only for 0.13, 0.09 and 0.05 respectively. We will proceed the research with the 50 audio features instead of reducing the dimensionality, since it would have a negative impact on the classification performance.

### 3.1.7. Final Variables and Quadrant Distribution

After the Pre-processing phase, the variables that will be used in the machine learning algorithms are shown in Table 4. As previously stated, one variable can have more than one 1st and 2nd statistics.

We will go into detail next on the models that will be created with these variables. It is important to remember that these variables do not have collinearity and can potentially impact the dependent variable.

When compared with the model used in the original paper using the DEAM Dataset, the audio features used are mostly the same (Aljanaki, 2016).

| Variable | Definition |
|---|---|
| ZeroCrossings | Counts the number of times the input signal crosses the zero line **(kurtosis)** |
| Rolloff | Frequency of sum of magnitudes of its lower frequencies are equal to percentage of the sum of magnitudes of its higher frequencies **(min)** |
| Flux | Normalized difference vector between two successive magnitude/power spectra **(mean, max, min)** |
| MFCC0 | Mel-Frequency cepstral coefficient 0 **(min)** |
| MFCC1 | Mel-Frequency cepstral coefficient 1 **(mean, skewness, min)** |
| MFCC2 | Mel-Frequency cepstral coefficient 2 **(skewness, kurtosis, max, min)** |
| MFCC3 | Mel-Frequency cepstral coefficient 3 **(mean, skewness, kurtosis)** |
| MFCC4 | Mel-Frequency cepstral coefficient 4 **(skewness, kurtosis)** |
| MFCC5 | Mel-Frequency cepstral coefficient 5 **(skewness, kurtosis, max, min)** |
| MFCC6 | Mel-Frequency cepstral coefficient 6 **(std, skewness, kurtosis)** |
| MFCC7 | Mel-Frequency cepstral coefficient 7 **(mean, skewness)** |
| MFCC8 | Mel-Frequency cepstral coefficient 8 **(skewness, kurtosis)** |
| MFCC9 | Mel-Frequency cepstral coefficient 9 **(mean, skewness, kurtosis)** |
| MFCC10 | Mel-Frequency cepstral coefficient 10 **(mean, skewness, kurtosis, max)** |
| MFCC11 | Mel-Frequency cepstral coefficient 11 **(mean, skewness, min)** |
| MFCC12 | Mel-Frequency cepstral coefficient 12 **(mean, skewness, kurtosis, max)** |
| PeakRatio_Chroma_D | Ratio of the highest peak to minimal/average peak of each observation in Chroma D from power spectogram **(skewness)** |
| PeakRatio_Chroma_D# | Ratio of the highest peak to minimal/average peak of each observation in Chroma D# from power spectogram **(std)** |
| PeakRatio_Average_Chroma_A | Ratio of the highest peak to average peak of each observation in Chroma A from power spectogram **(mean, skewness)** |
| PeakRatio_Minimum_Chroma_A | Ratio of the highest peak to minimal peak of each observation in Chroma A from power spectogram **(std, skewness)** |
| Rock | Dummy variable which defines if a song belongs to the Rock genre |
| Pop | Dummy variable which defines if a song belongs to the Pops genre |
| SoulRB | Dummy variable which defines if a song belongs to the SoulRB genre |
| Blues | Dummy variable which defines if a song belongs to the Blues genre |
| Electronic | Dummy variable which defines if a song belongs to the Electronic genre |
| Classical | Dummy variable which defines if a song belongs to the Classical genre |
| Hip-Hop | Dummy variable which defines if a song belongs to the Hip-Hop genre |
| International | Dummy variable which defines if a song belongs to the International genre |
| Experimental | Dummy variable which defines if a song belongs to the Experimental genre |
| Folk | Dummy variable which defines if a song belongs to the Folk genre |
| Jazz | Dummy variable which defines if a song belongs to the Jazz genre |
| Country | Dummy variable which defines if a song belongs to the Country genre |
| Artist | Artist that perfoms the song |
| Quadrants | Dependent variable which states in which Quadrant the song was annotated |

*Table 4 – Variables for the final model*

Since we removed outliers, the quadrant distribution is not exactly 200 for each of the quadrants, as shown in Figure 19. Quadrant 2 has more observations with 185 while Quadrant 3 has the least frequency with 176 songs. We consider that the difference is reasonable.

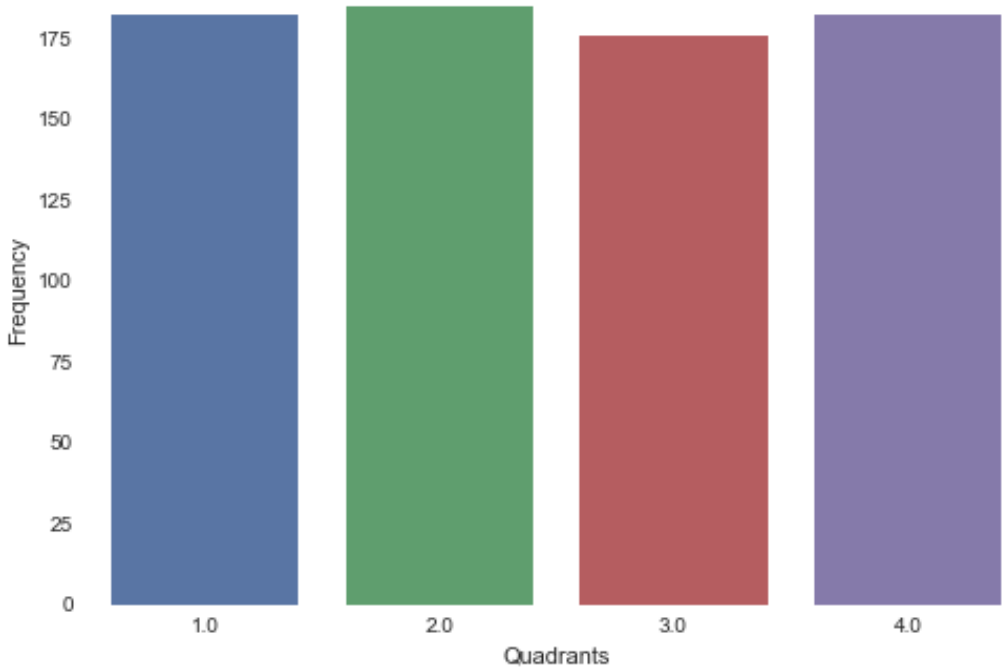



*Figure 19 - Quadrant distribution of songs*

## 3.2. Classification

Machine Learning algorithms learn from a body of data and generalize to new data that the algorithm had not encountered before. With the amount of data generated in today's world, the more useful Machine Learning algorithms are and more complex problems can be addressed by using them in the right way. Machine Learning is applied in several different areas such as ad placement, spam filtering and recommender systems. The most mature type of Machine Learning is classification (Domingos, 2012). There are several types of Machine Learning algorithms, but we will use the

supervised learning type in which the algorithm creates a function that maps the inputs to the outputs (Ayodele, 2010).

In our research, besides the focus to understand the role of Artist and Genre on the classification of songs by Emotion, we want to test different machine learning algorithms to understand which has the best performance. MER researchers have found that Support Vector Machines performs better when compared to other Machine Learning algorithms (Laurier & Herrera, 2007). We will compare this Machine Learning algorithm with others that are often used in MER: Naive Bayes, Decision Trees and K-Nearest Neighbour (Y.-H. Yang & Chen, 2012).

Support Vector Machines (SVM) is a supervised machine learning algorithm which can be used for regression or classification but it is mostly used for the latter. In an n-dimensional space, where n is the number of features of a model, each data point will be plotted and the classification will be performed by finding the hyper-plane that separates the classes. Support Vector Machines are the coordinates of individual observations which work has the frontier to separate two or more classes, also known as hyper-plane. This is a relatively straightforward procedure when the problem is separable, but when the data is not linearly separable, SVM uses kernels which are functions that transform the data points to separate them. This is helpful to convert non-linearly separable problems into separable problems. The main advantages of the SVM algorithm is the effectiveness in high dimensional spaces even if the dimensions surpass the number of samples, memory-efficiency and versatility. The main disadvantages are the fact that it does not provide probabilities to the estimations (Deng, Xu, & Li, 2010).

Naive Bayes (NB) is an algorithm used in Machine Learning that provides a way to calculate the probability of a hypothesis given prior knowledge. The assumption is that the input variables are independent, which can be assumed to be true in this case. The main advantages are intuitiveness, return of probabilities, not requiring large amounts of data and it is computationally fast (Ashari, Paryudi, & Tjoa, 2013). We used the Gaussian Naive Bayes which works with the Normal distribution. Since our input data is real-valued, we can safely assume that Gaussian Naive Bayes is the right choice (Lowd & Domingos, 2005).

Decision Trees are mostly used for classification problems. This algorithm splits the population into homogeneous subpopulations based on conditions in the input variables. Each decision tree is composed of nodes, where we split the data based on a variable; branches which are one side of the split; and leaves which is the result in the terminal nodes. We used the CART algorithm which uses the Gini coefficient to select criteria for maximum entropy between subpopulations. The algorithm selects the attribute with the smallest Gini coefficient to be the chosen criteria to perform the split. The main

advantages of Decision Trees are the easiness of interpretation, the fact that it is possible to work with both categorical and continuous input and output variables. The main disadvantage is the fact that, while it can create over-complex trees, it can lose generalization capacity easily (Ashari et al., 2013). Overfitting happens when the model cannot generalise what had been learned during the training phase to new data with potentially new characteristics. Therefore, there is a trade-off between performance and not creating excessively complex trees that cannot generalise to new data.

K-Nearest Neighbours (KNN) are widely used for pattern recognition problems. The easy output interpretation, fast calculation times and predictive power are regarded as advantages over other algorithms (Voulgaris & Magoulas, 2008). KNN labels a new data point by locating a number of nearest neighbours, which must be specified before the classification. To locate the nearest neighbours, we set KNN to use the Euclidean distance. When KNN locates the neighbours, if the number of neighbours' *k* was set to 1, the label of the new data point will be the label of the nearest neighbour. If *k*>1, we set all points in the neighbourhood to weight equally when labelling the new data point. One of KNN main disadvantages is being susceptible to noise in the training data such as outliers, which is one of the reasons why a considerable amount of time was spent Pre-processing the data (Imandoust & Bolandraftar, 2013). We started with a default *k*=5 but conducted a grid-search and selected the best performance number for *k.*

Firstly, we created different models to understand the impact of the combination of variables: Audio-only; Genre-only; Artist-only; Audio and Genre; Audio and Artist; Genre and Artist; Audio, Genre and Artist.

Secondly, we proceeded to the phase of feature selection for each of the former models. In Machine Learning, features do not have equal importance and, keeping irrelevant features may lead to inaccurate inferences. The goal of a Feature Selection Algorithm is to maximize the accuracy of the prediction while keeping the minimal dimensional size. The ReliefF algorithm is popular in MER due to its simplicity and effectiveness (Y.-H. Yang et al., 2008). This algorithm takes into account the feature interrelationship and assigns a rank to each of the features, which were evaluated one by one (Robnik-Šikonja & Kononenko, 2003). We used ten-fold cross-validation in the ReliefF algorithm. The software used to implement it was Weka[8]. Weka is an open-source software composed of a collection of machine learning algorithms developed by the University of Waikato for data mining tasks.

Following the feature rank given by the ReliefF algorithm, for each of the models, we used the Support Vector Machine algorithm with 20 times 10-fold stratified cross-validation and a greedy

[8] http://www.cs.waikato.ac.nz/ml/weka/

backwards feature selection approach to determine the optimal number of features. We used 20 times 10-fold stratified cross-validation because it has been shown by previous MER researchers to be the best method to analyse the performance of the models since it prevents overfitting, therefore helping to understand the generalization capacity of a classifier (Panda et al., 2015). The performance was analysed by the average f-measure. This metric is commonly used since it encapsulates the weighted average of the precision and recall metrics, as shown in Equation 12.

$$F - measure = 2 * \frac{precision * recall}{precision + recall}$$

*Equation 12 - F-measure metric*

Precision and Recall measures are commonly used to assess the quality of the predictions of classifications. Precision (Equation 13) is the ability of a classifier to correctly label a sample as positive. The values will vary between 0 and 1.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

*Equation 13 – Precision metric*

Recall (Equation 14) is the capacity of a classifier to find true positives. The values will vary between 0 and 1.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

*Equation 14 – Recall metric*

Afterwards, we used each model's features in the other machine learning models. When all the results were observed and compared, we started the classifier's optimization phase, in which we used grid-search to find the best performant parameters for the algorithms.

Subsequently, the difference between the results of each model combination of the best performant machine learning algorithm was tested for statistical significance. Since our models were trained and tested using the same population and seed, we used paired tests.

To compare these models, we firstly needed to check if the distribution of the fold's f-measures of each of the model is Gaussian by using the Komolgorov –Smirnov test. Under the null-hypotheses, the two distributions in the test are identical while the alternative hypothesis states that the distributions are not identical. We used a threshold for the p-value of 0.01 as it is normally used in MER researches (Panda et al., 2015). After finishing the test, since all the distributions follow a Normal distribution, we use the t-test to assess if the difference between the performance of the models combinations is statistically significant. We repeated the use of the threshold for the p-value of 0.01.

We then used the model with the best performance to classify the dataset. We used holdout validation consisting of 70% for the training set and 30% for the test set.

In the Results and Discussion, we will try different approaches such as the former mentioned 2nd approach to the Artist variable.

## 3.3. Visualization

To visualize what are the characteristics of the songs that were classified incorrectly, a dashboard was created using the Tableau[9] software. Tableau is a software developed to produce interactive data visualizations for business intelligence workers.

Our dashboard will be composed of three visualizations regarding misclassified songs: one focused on the distribution across the four quadrants; the genre distribution; and how many songs each artist composed in the misclassified sample.

The objective of the first two visualizations is straight-forward. The third visualization serves the purpose of understanding if the model classifies better artists which have more than one song in the dataset.

We followed the data-ink ratio guideline introduced by Edward Tufte which suggests that visualization designers should not include elements that do not deliver statistical information, which results in minimalistic graph designs (Tufte, 2001). Some authors defend that there is no evidence that high data-ink ratio visualizations result in increased graph comprehension but works as a guidance in the subjective world of visualization aesthetics (McGurgan, 2015). The data-ink ratio formula is shown in Equation 15.

[9] https://www.tableau.com/

$$Data - Ink\ Ratio = \frac{Data - Ink}{Total\ ink\ used\ in\ the\ Visualization}$$

*Equation 15 – Data-Ink Ratio*

To create a colour-blindness-friendly dashboard, we selected the Tableau's colour palette designed specifically to address this problem.

# 4. RESULTS AND DISCUSSION

## 4.1. Feature Selection

We created different models to inspect the impact of the combination of each group of variables: Audio-only; Genre-only; Artist-only; Audio and Genre; Audio and Artist; Genre and Artist; Audio, Genre and Artist.

To find the optimal number of features for the classification models, we used the rank retrieved by the ten-fold cross-validation ReliefF algorithm for each model and proceeded with a greedy backwards feature selection approach using SVM with 20 times 10-fold stratified cross-validation. As stated before, we assessed the performance of the models by observing the f-measure.

As shown in Figure 20, which is the representation of the F-measure results for the backwards feature selection approach for the model Audio, Genre and Artist, which has the audio features, artist and genre, the optimal number of features is 5. The optimal variables consist of *MFCC0(min), MFCC1(min), PeakRatio_Chroma_D#(std), Artist and International*.

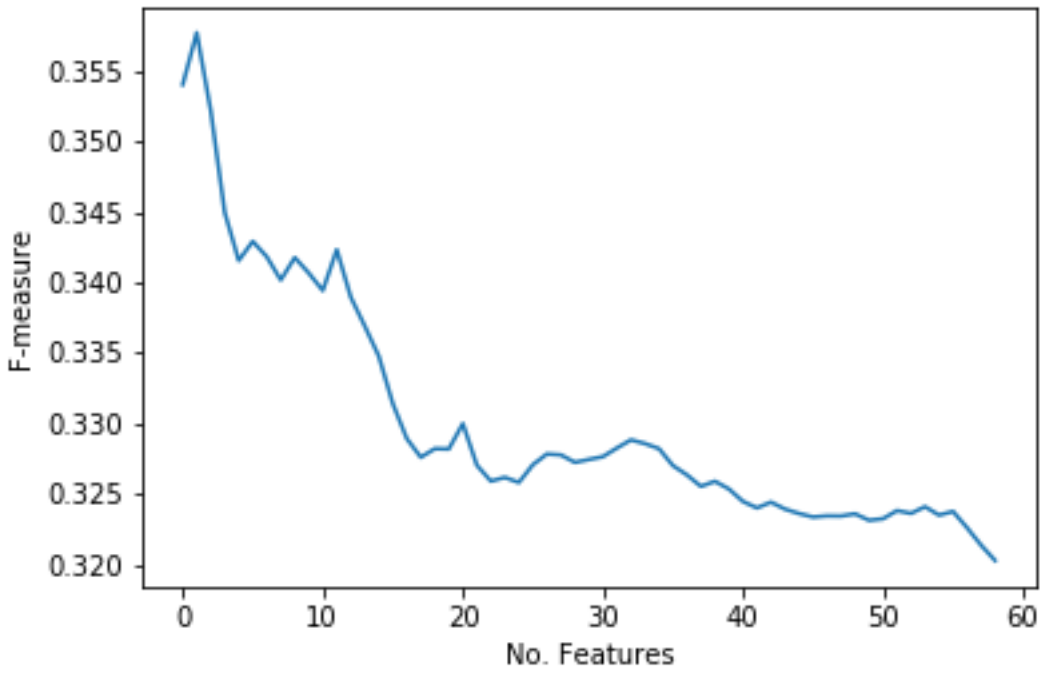



*Figure 20 – Backwards Feature Selection for Audio, Genre and Artist Model*

The Audio-only model, as shown in Appendix 1, has an optimal number of features of 21. The variables selected were *MFCC0(min), PeakRatio_Chroma_D#(std), PeakRatio_Chroma_D(skewness), MFCC1(mean), MFCC12(mean), MFCC1(min), MFCC2(min), MFCC2(max), MFCC12(kurtosis), Flux(mean), MFCC1(skewness), MFCC12(skewness), MFCC5(kurtosis), MFCC7(mean), MFCC2(skewness), Flux(max), MFCC4(kurtosis), PeakRatio_Minimum_Chroma_A(skewness), MFCC9(skewness), MFCC9(kurtosis)* and *MFCC5(max)*.

The Genre-only model has an optimal number of features of 12: *International, Pop, Hip-Hop, Blues, Rock, SoulRB, Folk, Electronic, Classical, Jazz, Country* and *Experimental*. This shows that the best performance was observed when all the genre variables were in the model as shown in Appendix 2.

The Artist-only model did not go through the backwards feature selection approach since the model is composed of only one variable.

The Audio and Genre model, as shown in Appendix 3, is optimally composed by a total of 21 variables: *PeakRatio_Chroma_D#(std), International, MFCC0(min), MFCC1(mean), MFCC1(min), PeakRatio_Minimum_Chroma_A(skewness), Hip-Hop, Rock, MFCC2(min), Jazz, MFCC12(max), MFCC2(max), PeakRatio_Chroma_D(skewness), MFCC1(skewness), MFCC7(mean), MFCC5(max)*, *Pop, MFCC8(kurtosis), MFCC6(std), Flux(mean), MFCC10(max).*

The Audio and Artist model consists of 9 variables, as shown in Appendix 4. Note that even if the F-measure is not the highest for this number of variables, it was the optimal number that included the variable *Artist*. The variables are *PeakRatio_Chroma_D#(std), MFCC0(min), MFCC1(mean), PeakRatio_Chroma_D(skewness), MFCC12(mean), MFCC2(max), MFCC2(min), MFCC11(min)* and *Artist*.

Lastly, the Genre and Artist model is composed of 5 variables: *Artist, International, Blues, Folk* and *Electronic* (Appendix 5). Regarding the optimal number of variables, the same principle of the Audio and Artist model was applied, which means that this the optimal F-measure that includes the variable *Artist*.

## 4.2. Classification Results

Having the optimal number of features for each of the SVM models, we compared the F-measure results of this machine learning algorithm with the same models with the same features but on the three other algorithms: Naive Bayes, Decision Trees and KNN. The 20 times ten-fold cross-validation was also employed on the other algorithms.

In the SVM algorithm, the default kernel used was the non-linear Radial Basis Function (RBF) and C and Gamma hyper-parameters were 1 and (1/number of features) respectively.

The Naive Bayes algorithm used was the Gaussian Naive Bayes as stated in the section before since our input data is real-valued. In the Decision Trees, the default parameters do not involve either a minimum number of observations to split nor a maximum number of levels. This is a simplified use of decision trees which will result in overfitting. This happens because the decision tree will create complex and specific rules which will lack the generalization capacity. We will address this problem later.

The KNN algorithm have 5 as the default number of neighbours and the neighbours voting is uniform, which results in the fact that all observations' vote in the neighbourhood is weighted equally.

On Table 5, it is possible to observe the comparison of the F-measure results across the 4 machine learning algorithms discussed before. The Audio, Genre and Artist model is named 'All'.

| | SVM | Naive Bayes | Decision Tree | KNN |
|---|---|---|---|---|
| Audio Only | 0.37 | 0.4 | 0.33 | 0.38 |
| Genre Only | 0.35 | 0.35 | 0.35 | 0.30 |
| Artist Only | 0.36 | 0.2 | 0.35 | 0.33 |
| Audio & Genre | 0.4 | 0.37 | 0.35 | 0.38 |
| Audio & Artist | 0.34 | 0.35 | 0.34 | 0.33 |
| Genre & Artist | 0.36 | 0.31 | 0.36 | 0.33 |
| All | 0.36 | 0.29 | 0.30 | 0.34 |

*Table 5 - F-measure results comparison before Grid-Search*

As it is easily observed, the Audio-only model has the best F-measure results in three of the four algorithms. In the Decision Tree results, since the decision trees were not pruned, the results from the Genre and Artist model can be due to overfitting. In the KNN algorithm, the Audio only model edges the Audio and Genre model since the unrounded F-measure results of these models is 0.3843 and 0.3782 respectively.

When analysing the results between the different machine learning algorithms, we conclude that the SVM has the best performance in more than 50% of the models and it is tied in first place on two of the three remaining models.

Even though the results of these models are low, they are still performing better than pure chance (25% probability) besides the Artist-only model of the Naive Bayes algorithm.

We decided to optimize the model's hyper-parameters of each of the machine learning algorithms to improve their performance.

## 4.3. Grid Search Improvements

Having to improve the results from the machine learning algorithms, we did a Grid-Search to find the right hyper-parameters for the algorithms.

On the SVM, we followed the Loose Grid Search which performs the search in C = $2^{-5}$, $2^{-3}$, $2^{-1}$,...,$2^{15}$ and Gamma = $2^{-15}$, $2^{-13}$, $2^{-11}$,...,$2^{3}$ (Hsu, Chang, & Lin, 2010). We also tried different kernels: Linear, RBF, Polynomial and Sigmoid.

As stated before, the results of the Decision Trees might have been inflated since one of the major disadvantages of this algorithm is that it easily falls into the overfitting scenario. To prevent this to happen, we had to prune the decision trees and at the same time maximizing the F-measure results for the models. The Grid-Search was performed on the minimum number of observations to perform a split between 2 and 500 records. The analysis extended to the maximum number of levels between 1 and 100 levels. These tweaks will prevent the model of overfitting because it will make harder the over complexity of the decision tree and subsequently lack of generalization capacity even if it implicates a worse performance of the algorithm.

The KNN algorithm was tested on the number of neighbours between 0 and 10. The grid-search is not applicable to the Naïve Bayes algorithm.

On Table 6, it is possible to observe the results between the models across the different machine learning algorithms after the Grid-Search improvements and check the respective improvement percentage. The Naive Bayes results kept the same since there were no improvements to the algorithm.

| | SVM | Increase (%) | Naive Bayes | Decision Tree | Increase (%) | KNN | Increase (%) |
|---|---|---|---|---|---|---|---|
| Audio Only | 0.46 | 24.32 | 0.4 | 0.36 | 9.09 | 0.41 | 7.90 |
| Genre Only | 0.37 | 5.71 | 0.35 | 0.35 | 0 | 0.30 | 0 |
| Artist Only | 0.36 | 0 | 0.2 | 0.36 | 2.86 | 0.35 | 6.06 |
| Audio & Genre | 0.45 | 12.5 | 0.37 | 0.37 | 5.71 | 0.41 | 7.90 |
| Audio & Artist | 0.40 | 17.65 | 0.35 | 0.36 | 5.88 | 0.38 | 15.15 |
| Genre & Artist | 0.38 | 5.56 | 0.31 | 0.32 | -11.11 | 0.35 | 6.06 |
| All | 0.38 | 5.56 | 0.29 | 0.30 | 0 | 0.37 | 8.82 |

*Table 6 - F-measure results comparison after Grid-Search improvements*

As it can be concluded, the SVM's performance is the best for all the models, which validates the conclusions taken in previous MER researches (Y.-H. Yang & Chen, 2012).

Regarding the improvements made by the Grid-Search, we highlighted the ones that had more than 10% of increase. The biggest improvements were observed on the SVM algorithm, especially in the Audio-only, Audio and Genre and Audio and Artist models in which the increase was 24.32%, 12.5% and 17.65% respectively. The KNN also registered a 15.15% increase on the Audio and Artist model.

Our suspicions were confirmed on the Genre and Artist model of the Decision Tree algorithm. While it was the algorithm that performed the best on the Genre and Artist model before the Grid-Search, now the performance decreased 11.11%. This happened because the model was overfitting and when we added a 56-minimum number of observations to perform the split, the model lost performance but gained generalization capacity.

When comparing the best performance between the models of the SVM algorithm, which again was the best performant algorithm across all models, Audio-only has the higher F-measure result in 0.46. Close to this result is the 0.45 Audio and Genre model. Before taking any conclusions from the results, we need to test the difference for statistical significance.

By taking the F-measures for each fold of each model of the SVM algorithm, which will from now on be the machine learning algorithm that will be our focus, since it performs the best, we will test the differences between each model for statistical significance. Since we used a 20 x ten-fold cross-validation, each model will be a vector of 200 F-measures.

Firstly, we had to test the Gaussianity of the distribution of each of the F-measure vectors. Since our models were trained and tested using the same population and seed, we used paired tests. To test Guassianity, we used the Komolgorov-Smirnov tests in which, in the null-hypotheses, the two distributions in the test are identical while the alternative hypothesis states that the distributions are not identical. We used a p-value threshold of 0.01 since it is commonly used in MER researches (Panda et al., 2015). We rejected the null hypothesis in all the models, thus concluding that the distribution of all the vectors is a Normal distribution.

Secondly, we used the t-test to assess if the difference in the model's results are statistically significant. The results led us to reject the null hypothesis, which stated that the results were due to chance and accepted the alternative hypothesis which states that the results have statistical significance. The threshold for the p-value used was the same as before, 0.01.

Now that we have the confirmation that the results are statistically significant, we can take informed conclusions.

As stated before, the Audio-only model has the best performance with an F-measure of 0.46 closely followed by the Audio and Genre model with 0.45 for F-measure. Both these models registered considerable increases when compared to before the Grid-Search Improvements. The highest increase appeared on the models that have audio features, so we can conclude that most of the improvements are observed on these group of variables. The C and Gamma of the Audio-only model is $2^{3}$ and $2^{-1}$ respectively, while for the Audio and Genre model is $2^{5}$ and $2^{-5}$ respectively. Both these models use a RBF kernel.

Even though the results of these models are above pure chance probability, which is 25%, it is not considered a good result. This not-so-good result can be due to the disadvantages of the DEAM dataset, which were analysed thoroughly before, including the lack of emotional stationary in the samples since the song excerpt is too long (45 seconds) and the lack of agreement between the songs' annotators.

The best models do not have the variable *Artist* as part of them, which is a confirmation of the Pearson's correlation coefficient test made earlier on the Pre-processing Phase which gave a coefficient of -0.04, which means that this variable does not have a considerable impact in the *Quadrants* variable. This can be due to the fact that, as analysed before, 74.86% of the Artists have only one song on the dataset. This leads the model to not have enough information on each artist to understand a pattern and therefore generalize it to new data.

To try to address the lack of repeating artists on the dataset, we will follow what we called in the Pre-processing phase the 2nd Approach to the Artist variable.

## 4.4. Artist Variable with the 2nd Approach

In the current approach to the Artist variable, in the case that an artist has a song with another artist, this combination will be considered as a new artist. We called this the 1st Approach to the Artist variable.

To improve the number of repeated artists in the dataset, we will follow the 2nd Approach, in which in the case that an Artist has a song with another Artist, the song will be attributed to the Artist which owns the song, e.g. the song is in the Artist's album.

On Figure 21, we see that when we follow the 2nd Approach, the total of Artists that only composed one song were 73.75%, which is a decrease of 1.1 percentage points. The representation of Artists that composed 2 and 3 songs increased in 1.5 and 0.4 percentage points respectively. This can be explained by the fact that some combinations of artists were being regarded as a new artist on the 1st Approach and on the 2nd Approach the song is being attributed to the main artist which had already other songs in the dataset. The maximum of songs recorded by one artist remained unaltered.

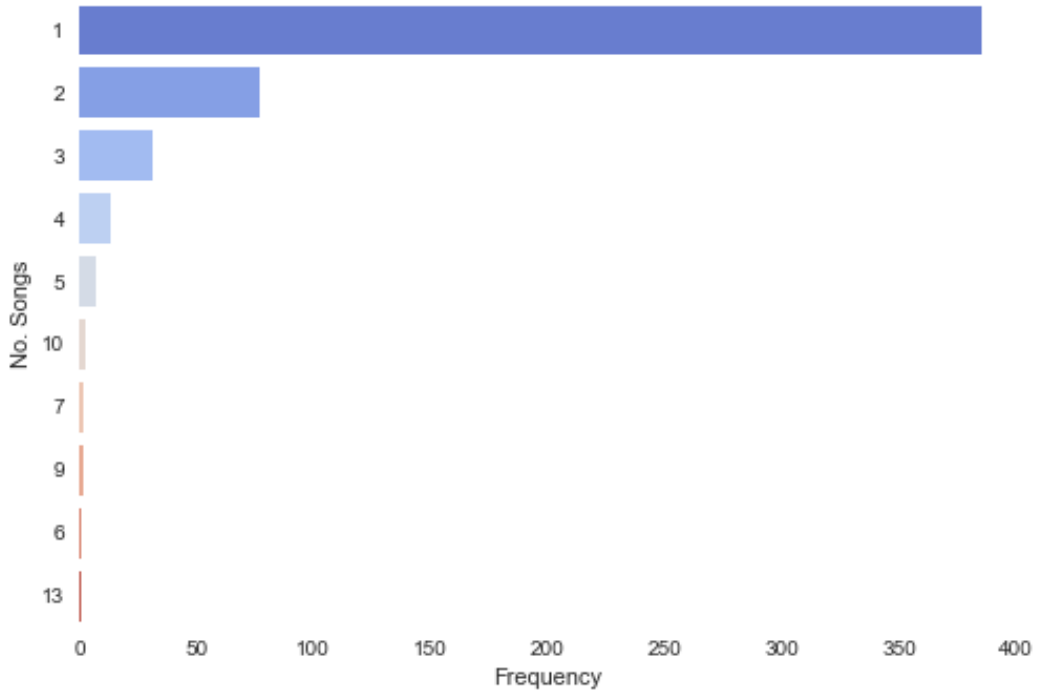



*Figure 21 - Number of Songs per Artist using 2nd Approach*

Now that we changed the variable, we ran the SVM algorithm to see if there are improvements in the models featuring the *Artist* variable.

For these models, we used the same features that we use in the original models and did a Grid-Search to find the optimal hyper-parameters for each of them. We will only compare with the models that have the Artist variable since it will be in these that the impact will be observed.

As shown in Table 7, there was almost no change to the previous results. It only changed in some millesimal places which rounded the result up or down. This is probably due to the fact that the number of one-time artists only decrease 1.1 percentage points.

| | 1st Approach | 2nd Approach |
|---|---|---|
| Audio Only | 0.46 | - |
| Genre Only | 0.37 | - |
| Artist Only | 0.36 | 0.34 |
| Audio & Genre | 0.45 | - |
| Audio & Artist | 0.40 | 0.41 |
| Genre & Artist | 0.38 | 0.37 |
| All | 0.38 | 0.37 |

*Table 7 - F-measure results comparison between Artist variable Approaches*

To understand if these results are statistically significant, we repeated the process of the Komolgorov-Smirnov test which resulted in the confirmation of the Gaussian distribution on the new model's F-measures. Afterwards, as done before, we did a t-test to compare the results of the 2nd Approach with the 1st Approach. With the results, we concluded that all the differences are statistically significant.

## 4.5. Classification using SVM

Now that we have the information on which are the best models and machine learning algorithm, we will use them to classify the songs. The main difference is that now we will use holdout validation method, in which we divide the dataset into training set (70%) and test set (30%).

Even though the Audio-only model is having the best performance, we will use the Audio and Genre model since one of the main objectives of this thesis is to understand the role of Artist and Genre. The *Artist* variable have been shown to not perform well but the Genre and Audio model have only a difference of F-measure of 0.0043 when compared with the Audio-only model. For this reason, we will continue using the Audio and Genre model with the SVM algorithm, which was the machine learning algorithm with the best performance.

The variables used for this model are: *PeakRatio_Chroma_D#(std), International, MFCC0(min), MFCC1(mean), MFCC1(min), PeakRatio_Minimum_Chroma_A(skewness), Hip-Hop, Rock, MFCC2(min),*

*Jazz, MFCC12(max), MFCC2(max), PeakRatio_Chroma_D(skewness), MFCC1(skewness), MFCC7(mean), MFCC5(max), Pop, MFCC8(kurtosis), MFCC6(std), Flux(mean), MFCC10(max).*

We did a Grid-Search to find which are the best hyper-parameters of the SVM and the results is an RBF kernel and C and Gamma of $2^{5}$ and $2^{-5}$ respectively.

In the Table 8, it is possible to observe the results for the test set and training set. We added the training set results to spot if there is overfitting in the model.

| | **No. Records** | **F-measure** | **Precision** | **Recall** |
|---|---|---|---|---|
| Training Set | 725 | 0.52 | 0.53 | 0.53 |
| Test Set | 218 | 0.47 | 0.48 | 0.49 |

*Table 8 – Results of Training and Test sets on the Audio and Genre model*

We can conclude that the model is not overfitting, thus not having to tweak the SVM hyper-parameters to improve generalization. This is good since we would lose performance if we had to improve generalization.

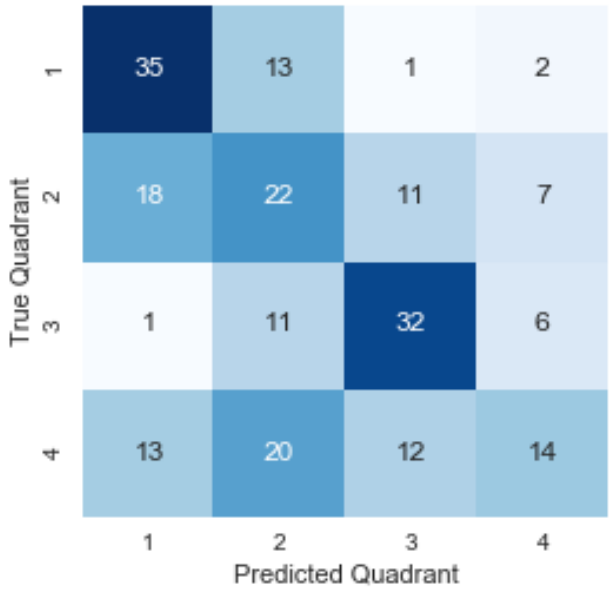



*Figure 22 – Model's Confusion Matrix*

On Figure 22, we can investigate the confusion matrix resulting from the model predictions. It is possible to observe that the model is classifying correctly more times the Quadrants 1 and 3.

In this testing set, there were less songs classified for the Quadrant 4 when compared to the other quadrants with only 29 songs. Quadrants 1 and 2 have nearly the same number of predictions with 67 and 66 songs respectively. Quadrant 3 is close with 56 predictions. This means that the Quadrant with the best percentage of correctly classified songs is Quadrant 3 with 57,14%. On the contrary, Quadrant 2 is the most incorrectly predicted with 33,33%. This means that our model is best at predicting Melancholic songs (Quadrant 3) than Tense songs (Quadrant 2).

## 4.6. Analysis of Incorrectly Classified Songs

To understand which are the main characteristics of the misclassified songs, we will do a deeper analysis on these samples.

When we analysed the DEAM dataset, we created an Agreement Rate, which was the percentage of annotators that agreed with the Quadrant annotated in the dataset. Since our model's F-measure is 0.47, which is not considered a high performant model, it is possible that the reason for this result lies in the annotations. As concluded in the DEAM dataset analysis, 67.40% of the songs in the total dataset had an Agreement Rate of less than 51%. This means that the songs are not straightforward in terms of which of the quadrant they should be in. From the misclassified songs of our Audio and Genre model, it is observable that 93 of the 115 are samples that had an Agreement of less than 51%. This means that 80.87% of the incorrectly classified songs are high disagreement songs between the annotators. This is higher than their representation on the total dataset. With this data, we can conclude that the disadvantages of these dataset are bringing our performance down.

As stated in the DEAM dataset analysis, this can happen because the samples are 45-seconds, which is too long for static annotations. By being longer than they should, the samples can have more than one emotion in the sample, losing emotional stationarity.

Another reason for this low result can be the audio features which are not the most advanced. Therefore, the model is not catching meaningful characteristics of the songs thus classifying it wrongly.

Repeating what we have done in the DEAM dataset analysis, we took a small sample of 20 songs and classified them in the quadrants and saw if our interpretation is like the one of the annotators in our model's misclassified songs. This time we did the same process and had a 15% agreement with the annotations which is much different than the 80% of agreement we had when we did this procedure in the DEAM dataset analysis.

It is important to state that when we did the last procedure, we found a song in our sample which was a 45-second sample of an intro of a song in which a person was talking without any music. This was one of the songs that had low Agreement Rate.

## 4.6. Dashboard

To visualize the incorrectly classified songs, we created a dashboard that will be a good tool to understand the main characteristics of this set of songs. The dashboard was created using Tableau software.

Our dashboard is composed of three visualizations regarding misclassified songs: one focused on the distribution across the four quadrants; the genre distribution; and how many songs the artists that composed the misclassified samples have in the database, as shown on Figure 23. It can be found online[10] and interacted with.

In the "True Quadrant Distribution", the size of the circles is proportional to the number of misclassified songs that were labelled as one of these quadrants.

In the "Genre Distribution", we can visualize which are the genres that have more representation on this set of incorrectly classified songs. This can help understand what are the most problematic genres. In this case, the Genre representation as a percentage follows the same line as the representation in the whole dataset.

In the last visualization, "Songs by Artists", it is observable which are the most represented artists in this set of songs. Jason Shaw is the Artist with the most songs in these incorrectly classified songs list but he is also the artist with the highest number of songs in the dataset. The curious part is Fit and the Conniptions being the artists with the second most songs in this dataset when they are not on the top 5 most represented artists in the whole dataset. This means that 43% of their songs has been misclassified.

As stated before, we followed the data-ink ratio guidelines to avoid visual noise in the dashboard. All the visualizations are colour-blindness-friendly since we used Tableau's colour palette designed specifically to address this problem.

[10] https://public.tableau.com/views/Thesis_8/Dashboard1?:embed=y&:display_count=yes

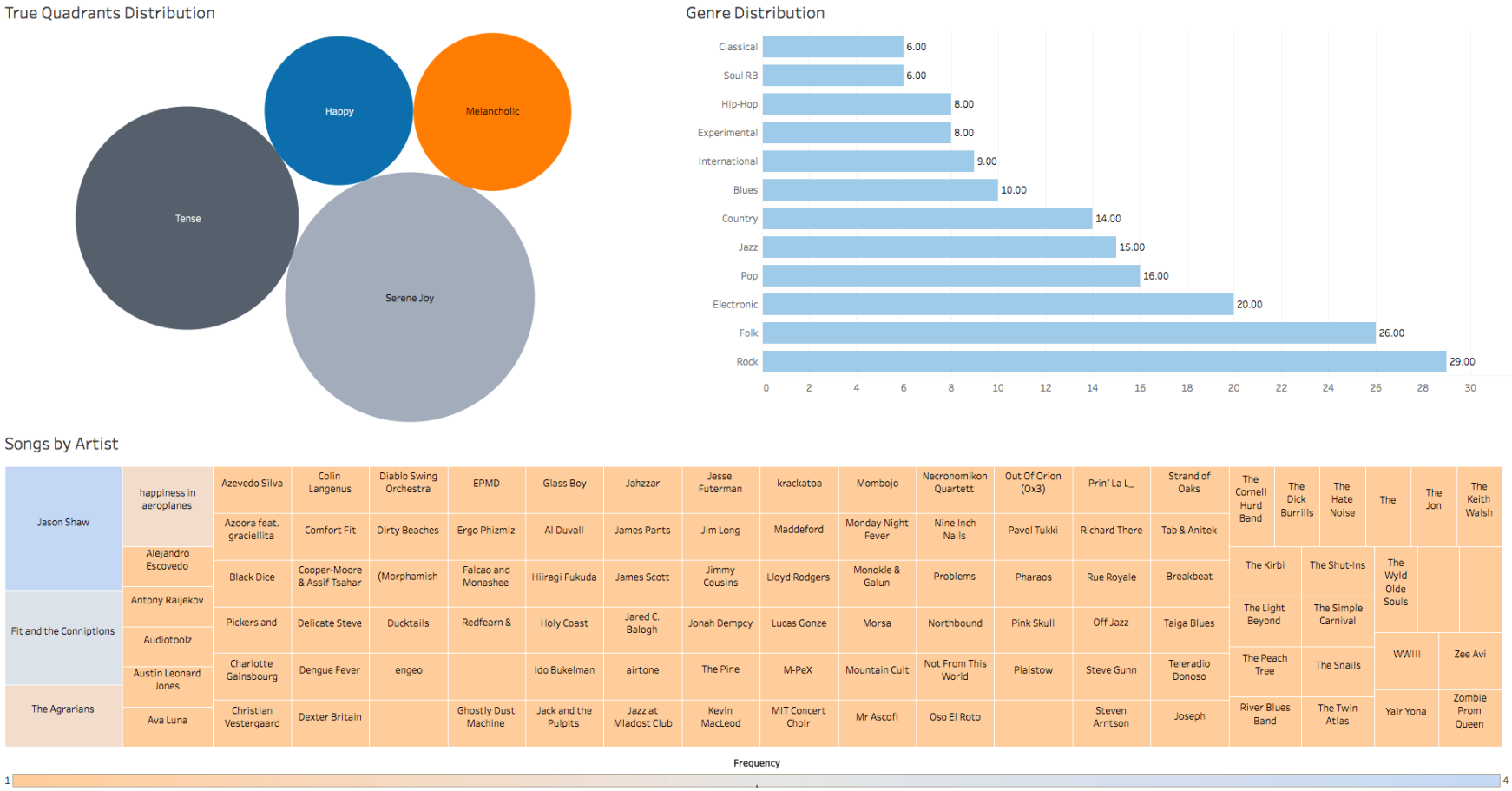



*Figure 23 - Dashboard of Misclassified songs*

# 5. CONCLUSIONS AND FUTURE WORK

At the end of the thesis we have the strong belief that all the objectives were delivered and, although we found several fragilities on the data in the DEAM dataset, we were still able to deliver the best conclusions regarding scientific rigor.

After the conclusion of the research, the impact of Artist and Genre on the Emotion of the songs was comprehended and various approaches were taken in order to deepen our knowledge about these variables when combined with audio features.

We also validated previous researches regarding the fact that the Support Vector Machines algorithm is the best performant when compared with other commonly used algorithms in MER such as Decision Trees, Naive Bayes and K-Nearest Neighbors.

The dashboard created brought a new approach to the research by analyzing the incorrectly classified songs and understand their characteristics regarding emotion Quadrant, Genre and Artist.

Two meaningful contributions were the fact that we did not find researches using the DEAM dataset for classification regarding static annotations nor we found any researches using the Artist variable in MER.

## 5.1. Future Work

We propose that future researches use a predicted genre to create the classification models to add a realistic error layer to the problem. We used annotated genres which can be subjective and hard to quantify the error.

Furthermore, we recommend the use of a dataset with a higher annotation Agreement Rate. In our opinion, it is also beneficial to use a dataset which has a larger representation of each artist and genre and a balance between Quadrants.

Lastly, we propose the use of more advanced audio features and more adequate for artists' classification (e.g. Artist's Voice Timbre), to reach a higher performance in the classification model.

# 6. BIBLIOGRAPHY


Aljanaki, A. (2016). *Emotion in Music: representation and computational modeling*. Utrecht University.

Aljanaki, A., Wiering, F., & Veltkamp, R. C. (2016). Studying emotion induced by music through a crowdsourcing game. *Information Processing and Management*, *52*(1), 115–128. https://doi.org/10.1016/j.ipm.2015.03.004

Aljanaki, A., Yang, Y., & Soleymani, M. (2016). Benchmarking music emotion recognition systems. *PLoS ONE*.

Amado-Boccara, I., Donnet, D., & Olié, J. P. (1993). Conception of mood in psychology. *Encephale*, *19*(2), 117–122.

Ashari, A., Paryudi, I., & Tjoa, A. M. (2013). Performance Comparison between Naïve Bayes , Decision Tree and k-Nearest Neighbor in Searching Alternative Design in an Energy Simulation Tool. *International Journal of Advanced Computer Science and Applications*, *4*(11).

Ayodele, T. O. (2010). Types of Machine Learning Algorithms. In *Types of Machine Learning Algorithms, New Advances in Machine Learning*. https://doi.org/10.5772/9385

Bischoff, K., Firan, C. S., Paiu, R., Nejdl, W., Laurier, C., & Sordo, M. (2009). Music Mood and Theme Classification - a Hybrid Approach. In *Proceedings of the 10th International Society for Music Information Retrieval Conference* (pp. 657–662).

Calder, A. J., Lawrence, A. D., & Young, A. W. (2001). Neuropsychology of Fear and Loathing. *Nature Reviews Neuroscience*, *2*(5), 352–363. https://doi.org/10.1038/35072584

Chen, Y., Yang, Y., Wang, J., & Chen, H. (2015). The AMG1608 Dataset for Music Emotion Recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/ICASSP.2015.7178058

Coutinho, E., & Scherer, K. R. (2012). Towards a brief domain-specific self-report scale for the rapid assessment of musically induced emotions. In *Proceedings of the 12th International Conference on Music Perception and Cognition and the 8th Triennial Conference of the European Society for the Cognitive Sciences of Music*.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of test. *Psychometrika*, *16*, 297–334.

Deng, C., Xu, L., & Li, S. (2010). Classification of support vector machine and regression algorithm. https://doi.org/10.5772/9392

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, *55*(10), 78–87. https://doi.org/10.1145/2347736.2347755

Dormann, C. F., Elith, J., Bacher, S., Buchmann, C., Carl, G., Carré, G., … Lautenbach, S. (2013). Collinearity: A review of methods to deal with it and a simulation study evaluating their performance. *Ecography*, *36*(1), 27–46. https://doi.org/10.1111/j.1600-0587.2012.07348.x

Downie, J. S. (2008). The music information retrieval evaluation eXchange (2005-2007): A window into music information retrieval research. *Acoustical Science and Technology*, *29*(4), 247–255. https://doi.org/10.1250/ast.29.247

Eerola, T., & Vuoskoski, J. K. (2011). A comparison of the discrete and dimensional models of emotion in music. *Psychology of Music*, *39*(1), 18–49. https://doi.org/10.1177/0305735610362821

Ekman, P. (1992). An argument for basic emotions. *Cognition & Emotion*. https://doi.org/10.1080/02699939208411068

Ellis-Petersen, H. (2016). Streaming growth helps digital music revenues surpass physical sales. Retrieved from https://www.theguardian.com/music/2016/apr/12/streaming-revenues-bring-big-boost-to-global-music-industry

Farnsworth, P. R. (1954). A Study Of The Hevner Adjective List. *The Journal of Aesthetics and Art Criticism*, *13*(1), 97–103. https://doi.org/10.2307/427021

Feng, Y., Zhuang, Y., & Pan, Y. (2003). Music information retrieval by detecting mood via computational media aesthetics. In *IEEE/WIC International Conference on Web Intelligence*. https://doi.org/10.1109/WI.2003.1241199

Gabrielsson, A. (2001). Emotion perceived and emotion felt: Same or different? *Musicae Scientiae*.

Gabrielsson, A., & Juslin, P. N. (1996). Emotional Expression in Music Performance: Between the Performer's Intention and the Listener's Experience. *Psychology of Music*, *24*(1), 68–91. https://doi.org/10.1177/0305735696241007

Hauke, J., & Kossowski, T. (2011). Comparison of Values of Pearson's and Spearman's Correlation Coefficients on the Same Sets of Data. *Quaestiones Geographicae*, *30*(2), 87–93. https://doi.org/10.2478/v10117-011-0021-1

Hevner, K. (1935). Expression in music: a discussion of experimental studies and theories. *Psychological Review*, *42*(2), 186–204. https://doi.org/10.1037/h0054832

Hsu, C., Chang, C., & Lin, C. (2010). A Practical Guide to Support Vector Classification.

Hu, X., & Downie, J. S. (2007). Exploring Mood Metadata: Relationships with Genre, Artist and Usage Metadata. In *Proceedings of the 8th International Conference on Music Information Retrieval* (pp. 67–72).

Hu, X., & Downie, J. S. (2010). When Lyrics Outperform Audio for Music Mood Classification: A Feature Analysis. In *Proceedings of the 11th International Society for Music Information Retrieval Conference*.

Hu, X., & Liu, J. (2010). Evaluation of Music Information Retrieval: Towards a User-Centered Approach.

Imandoust, S. B., & Bolandraftar, M. (2013). Application of K-Nearest Neighbor ( KNN ) Approach for Predicting Economic Events : Theoretical Background. *Journal of Engineering Research and Applications*, *3*(5), 605–610.

Kim, Y. E., Schmidt, E. M., Migneco, R., Morton, B. G., Richardson, P., Scott, J., … Turnbull, D. (2010). Music Emotion Recognition : a State of the Art Review. In *11th International Society for Music Information and Retrieval Conference*.

Kleinginna, P. R., & Kleinginna, A. M. (1981). A categorized list of emotion definitions, with suggestions for a consensual definition. *Motivation and Emotion*, *5*(4), 345–379. https://doi.org/10.1007/BF00992553

Lartillot, O. (2009). *MIRtoolbox 1.2 User's Manual*.

Lartillot, O. (2013). *MIRtoolbox 1.4 User's Manual*.

Lartillot, O., & Toiviainen, P. (2007). A matlab toolbox for musical feature extraction from audio. In *Proc. of the 10th International Conference on Digital Audio Effects*.

Laurier, C. (2011). *Automatic Classification of Musical Mood by Content-Based Analysis*.

Laurier, C., Grivolla, J., & Herrera, P. (2008). Multimodal music mood classification using audio and lyrics. In *Proceedings - 7th International Conference on Machine Learning and Applications*. https://doi.org/10.1109/ICMLA.2008.96

Laurier, C., & Herrera, P. (2007). Audio music mood classification using support vector machine. In *International Society for Music Information Retrieval Conference (ISMIR)*.

Lee, K. (2008). A system for automatic chord transcription from audio using genre-specific hidden Markov models. In *Adaptive Multimedia Retrieval: Retrieval, User, and Semantics*. https://doi.org/10.1007/978-3-540-79860-6_11

Li, T., & Ogihara, M. (2003). Detecting emotion in music. *Proceedings of the International Symposium on Music Information Retrieval*.

Lin, Y., Chen, X., & Yang, D. (2013). Exploration of Music Emotion Recognition Based on MIDI.

Lin, Y., Yang, Y., Chen, H. H., Liao, I., & Ho, Y. (2009). Exploiting genre for music emotion classification. In *Proceedings of the 2009 IEEE International Conference on Multimedia and Expo* (pp. 618–621).

Liu, A., Ghosh, J., & Martin, C. E. (2007). Generative Oversampling for Mining Imbalanced Datasets. In *Proceedings of the 2007 International Conference on Data Mining*.

Liu, D., Lu, L., & Zhang, H.-J. (2003). Automatic mood detection from acoustic music data. In *Proceedings of the International Conference on Music Information Retrieval*.

Lowd, D., & Domingos, P. (2005). Naive Bayes Models for Probability Estimation. In *Proceedings of the Twenty-Second International Conference*. https://doi.org/10.1145/1102351.1102418

Lu, Q., Chen, X., Yang, D., & Wang, J. (2010). Boosting for Multi-Modal Music Emotion Classification. In *Proceedings of the 11th International Society for Music Information Retrieval Conference*.

Malheiro, R. (2016). *Emotion-based Analysis and Classification of Music Lyrics*.

Malheiro, R., Oliveira, H. G., Gomes, P., & Paiva, R. P. (2016). Keyword-Based Approach for Lyrics Emotion Variation Detection. In *8th International Conference on Knowledge Discovery and*

*Information Retrieval*. https://doi.org/10.5220/0006037300330044

McEnnis, D., McKay, C., & Fujinaga, I. (2005). jAudio: A feature extraction library. In *Proceedings of the International Conference on Music Information Retrieval*.

McFee, B., Raffel, C., Liang, D., Ellis, D. P. W., McVicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and Music Signal Analysis in Python. In *Proceedings of the 14th Python in Science Conference*.

McGurgan, K. (2015). *Data-ink Ratio and Task Complexity in Graph Comprehension*.

McKay, C. (2005). jAudio: Towards a standardized extensible audio music feature extraction system.

Mela, C. F., & Kopalle, P. K. (2002). The impact of collinearity on regression analysis: the asymmetric effect of negative and positive correlations. *Applied Economics*, *34*(6), 667–677. https://doi.org/10.1080/00036840110058482

Moffat, D., Ronan, D., & Reiss, J. (2015). An Evaluation of Audio Feature Extraction Toolboxes. In *Proceedings of the 18th International Conference on Digital Audio Effects*. https://doi.org/10.13140/RG.2.1.1471.4640

Ortony, A., & Turner, T. J. (1990). What's basic about basic emotions? *Psychological Review*, *97*(3), 315–331. https://doi.org/10.1037/0033-295X.97.3.315

Panda, R. (2010). Automatic Mood Tracking in Audio Music. *Audio Engineering*.

Panda, R., Malheiro, R., Rocha, B., Oliveira, A., & Paiva, R. P. (2013). Multi-Modal Music Emotion Recognition: A New Dataset, Methodology and Comparative Analysis. In *Proceedings of 10th International Symposium on Computer Music Multidisciplinary Research*.

Panda, R., & Paiva, R. P. (2011). Automatic Creation of Mood Playlists in the Thayer Plane: A Methodology and a Comparative Study. In *8th Sound and Music Computing Conference*.

Panda, R., Rocha, B., & Paiva, R. P. (2015). Music Emotion Recognition with Standard and Melodic Audio Features. *Applied Artificial Intelligence*, *29*(4), 313–334.

Pasick, A. (2015). The magic that makes Spotify's Discover Weekly playlists so damn good. Retrieved from https://qz.com/571007/the-magic-that-makes-spotifys-discover-weekly-playlists-so-damn-good/

Patro, S. G. K., & Sahu, K. K. (2015). Normalization: A Preprocessing Stage. https://doi.org/10.17148/IARJSET.2015.2305

Robnik-Šikonja, M., & Kononenko, I. (2003). Theoretical and Empirical Analysis of ReliefF and RReliefF. In *Machine Learning* (Vol. 53, pp. 23–69). https://doi.org/10.1023/A:1025667309714

Russell, J. A. (1980). A Circumplex Model of Affect. *Journal of Personality and Social Psychology*, *39*(6), 1161–1178. https://doi.org/10.1037/h0077714

Saari, P., & Eerola, T. (2014). Semantic Computing of Moods Based on Tags in Social Media of Music. *IEEE Transactions on Knowledge and Data Engineering*, *26*(10), 2548–2560.

https://doi.org/10.1109/TKDE.2013.128

Scherer, K. R., & Zentner, M. R. (2001). Emotional effects of music: Production rules. In *Music and emotion: Theory and research*.

Schmidt, E., & Kim, Y. (2011). Modeling Musical Emotion Dynamics With Conditional Random Fields. In *12th International Society for Music Information Retrieval Conference*.

Seo, S. (2006). A review and comparison of methods for detecting outliers in univariate data sets.

Soleymani, M., Aljanaki, A., & Yang, Y. (2016). *DEAM : MediaEval Database for Emotional Analysis in Music*.

Song, Y., Dixon, S., & Pearce, M. (2012). Evaluation of Musical Features for Emotion Classification. In *Proceedings of 13th International Society for Music Information Retrieval Conference (ISMIR)*.

Steinbach, M., Ertöz, L., & Kumar, V. (2003). The Challenges of Clustering High Dimensional Data. https://doi.org/10.1007/978-3-662-08968-2_16

Tellegen, A., & Clark, L. A. (1999). On the Dimensional and Hierarchical Structure of Affect. *Psychological Science*, *10*(4). https://doi.org/10.1111/1467-9280.00157

Trohidis, K., Tsoumakas, G., Kalliris, G., & Vlahavas, I. (2011). Multi-label classification of music by emotion. In *EURASIP Journal on Audio, Speech and Music Processing*. https://doi.org/10.1186/1687-4722-2011-426793

Tufte, E. (2001). *The visual display of quantitative information*. *Cheshire*.

Tzanetakis, G. (2002). *Manipulation, analysis and retrieval systems for audio signals*.

Van Der Maaten, L. J. P., Postma, E. O., & Van Den Herik, H. J. (2009). Dimensionality Reduction: A Comparative Review.

Voulgaris, Z., & Magoulas, G. D. (2008). Extensions of the k Nearest Neighbour Methods for Classification Problems.

Wager, T. D., Barrett, L. F., Bliss-Moreau, E., Lindquist, K. A., Duncan, S., Kober, H., … Mize, J. (2008). The Neuroimaging of Emotion. In *Handbook of Emotions* (pp. 249–271).

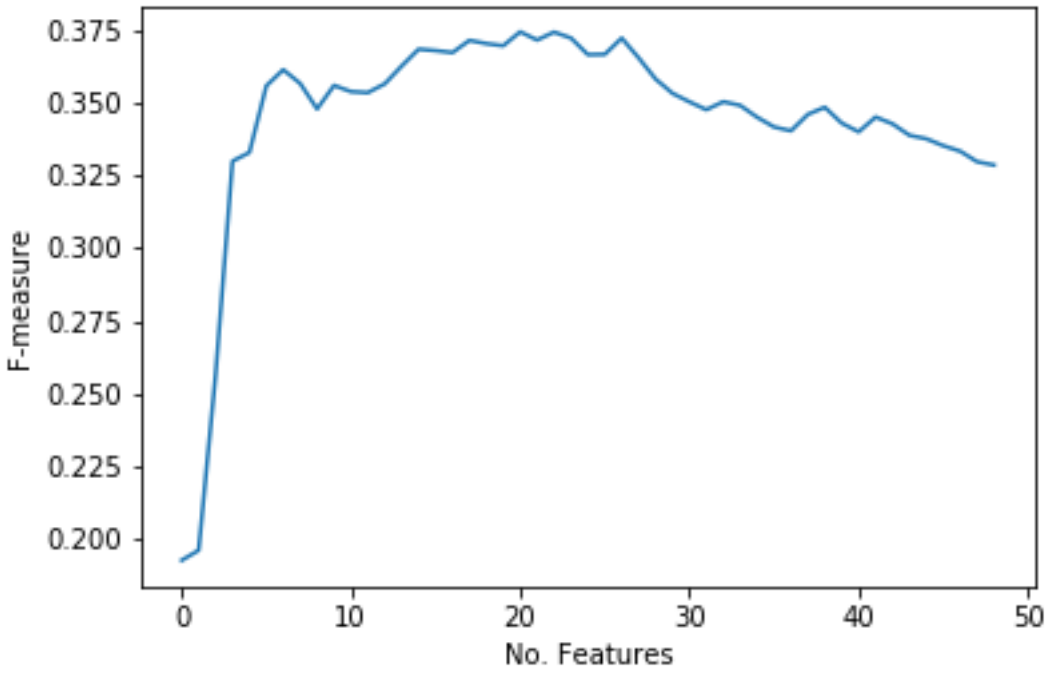
Yang, D., & Lee, W. (2004). Disambiguating Music Emotion Using Software Agents. In *Proceeding of 5th International Conference on Music Information Retrieval*.

Yang, Y.-H., & Chen, H. H. (2012). Machine Recognition of Music Emotion. *ACM Transactions on Intelligent Systems and Technology*, *3*(3). https://doi.org/10.1145/2168752.2168754

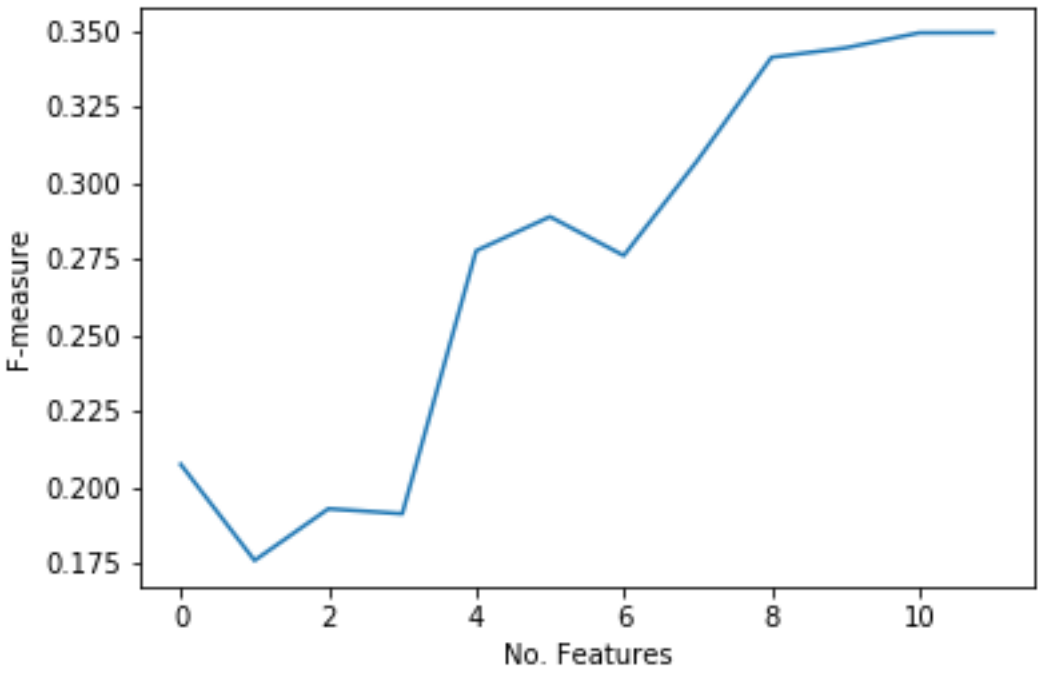
Yang, Y.-H., Lin, Y.-C., Su, Y.-F., & Chen, H. H. (2008). A Regression Approach to Music Emotion Recognition. *IEEE Transactions on Audio, Speech and Language Processing*, *16*(2), 448–457. https://doi.org/10.1109/TASL.2007.911513

Zentner, M., Grandjean, D., & Scherer, K. R. (2008). Emotions evoked by the sound of music: characterization, classification, and measurement. *Emotion*, *8*(4), 494–521. https://doi.org/10.1037/1528-3542.8.4.494

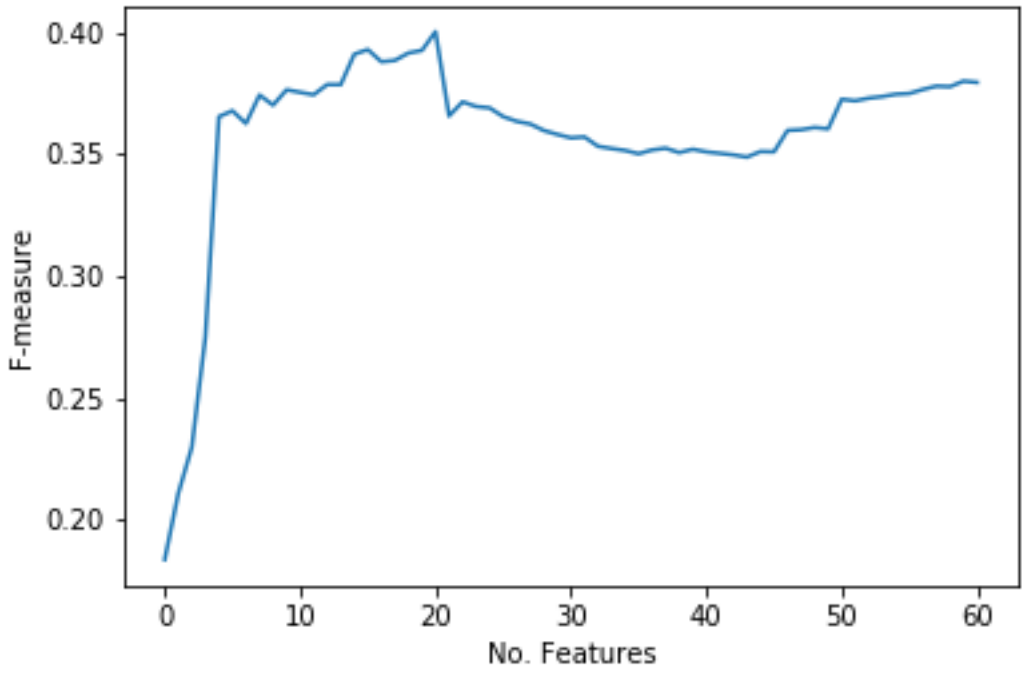
# 7. APPENDIX



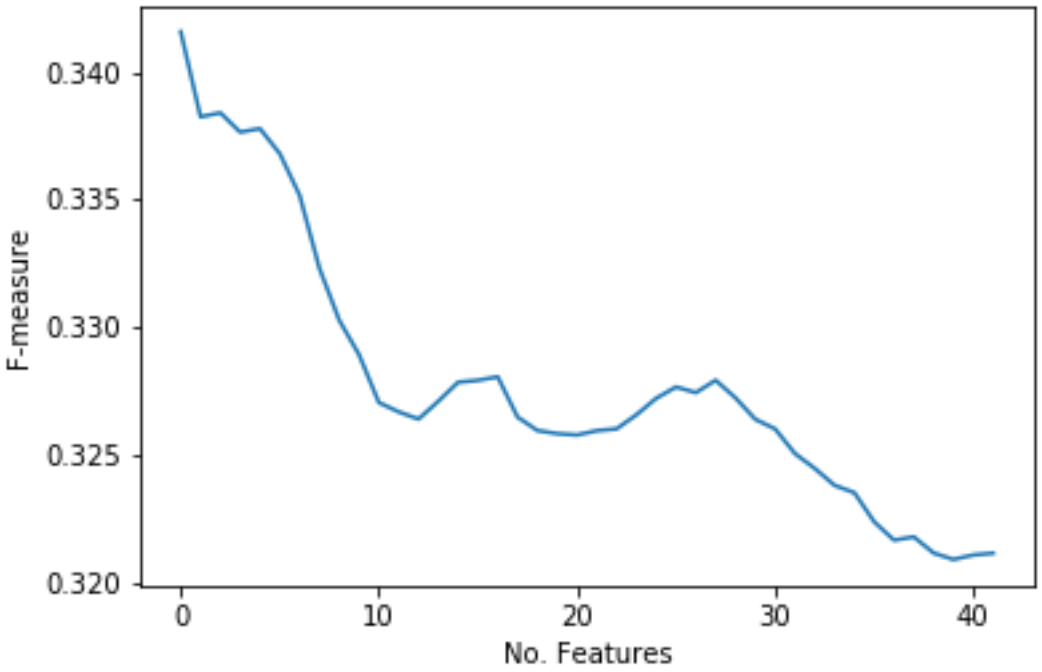

*Appendix 1 - Backwards Feature Selection for Audio-only Model*



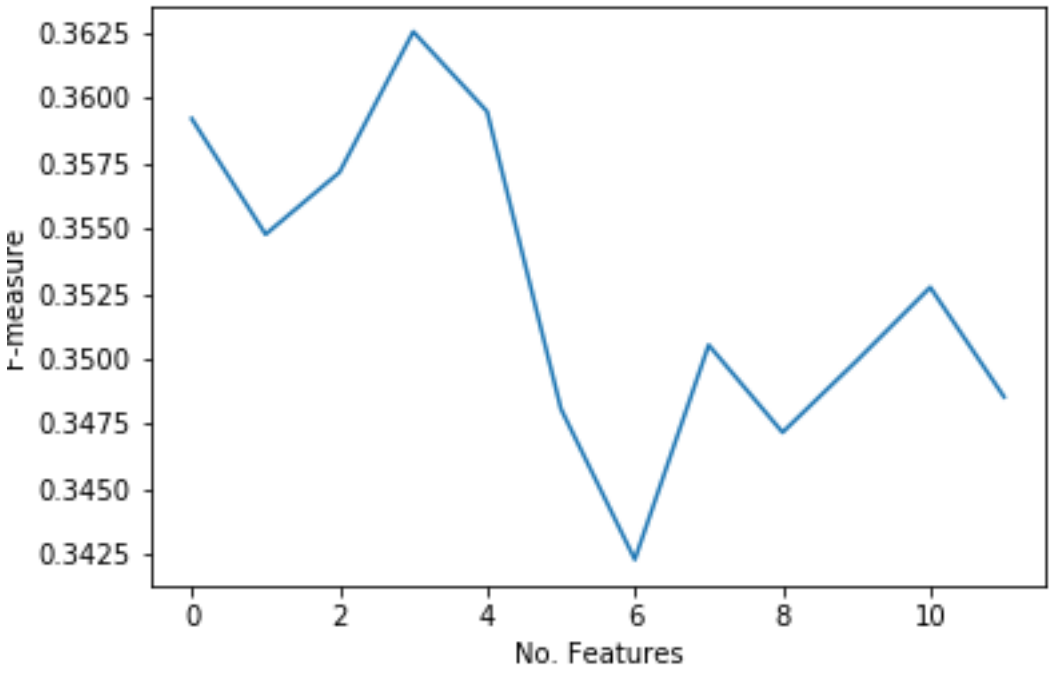

*Appendix 2 - Backwards Feature Selection for Genre-only Model*




*Appendix 3 - Backwards Feature Selection for Audio and Genre Model*




*Appendix 4 - Backwards Feature Selection for Audio and Artist Model*




*Appendix 5 - Backwards Feature Selection for Genre and Artist Model*