EXPLORING MUSICAL RELATIONS USING ASSOCIATION RULE NETWORKS

Renan de Padua[1,2] **Verônica Oliveira de Carvalho**[3]
Solange de Oliveira Rezende[1] **Diego Furtado Silva**[4]
[1] Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo, São Carlos, Brazil
[2] Data Science Team, Itaú-Unibanco, São Paulo, Brazil*
[3] Instituto de Geociências e Ciências Exatas – Universidade Estadual Paulista, Rio Claro, Brazil
[4] Departamento de Computação – Universidade Federal de São Carlos, São Carlos, Brazil
padua@icmc.usp.br, veronica@rc.unesp.br, solange@icmc.usp.br, diegofs@ufscar.br

ABSTRACT

Music information retrieval (MIR) has been gaining increasing attention in both industry and academia. While many algorithms for MIR rely on assessing feature subsequences, the user normally has no resources to interpret the significance of these patterns. Interpreting the relations between these temporal patterns and some aspects of the assessed songs can help understanding not only some algorithms' outcomes but the kind of patterns which better defines a set of similarly labeled recordings. In this work, we present a novel method to assess these relations, constructing an association rule network from temporal patterns obtained by a simple quantization process. With an empirical evaluation, we illustrate how we can use our method to explore these relations in a varied set of data and labels.

1. INTRODUCTION

Digital music repositories and streaming music services have become increasingly popular in the last decades. Along with this growth, algorithms to automatically organize, navigate, and search on music collections are more and more necessary. For this reason, Music information retrieval (MIR) has been gaining considerable attention in both industry and academia.

There is a multitude of MIR methods that rely on assessing subsequences of features. In other words, these methods extract features from the audio in a sliding window fashion and use successive subsets of these features to take decisions over the data. One example is the music genre classification, in which a common approach is to aggregate features from subsequences to obtain a more robust set of features [2, 8]. Moreover, Silva et al. [12] showed how assessing distances between subsequences can be used as a subroutine for different MIR tasks, from cover recognition to visualization.

In this work, we propose the use of a novel category of association rules networks to support understanding the relations between sequential patterns and labels which describe our data. The exploration of these association rules may provide insights on what kind of pattern defines one label, which may have implications on musicology or other MIR tasks.

For instance, consider the genre as the target label. If one pattern (or a group of patterns) happens with high confidence for only one label, this may help to explain the characteristics which define that genre. Also, it may guide us to understand how to improve music classification algorithms. Besides, our method helps us to find patterns shared between different labels. This kind of relation can be used, for example, to improve music recommendation systems, as well as provides insights on the musical influences between different labels.

Figure 1 illustrates one example of relations found by our method. It represents that, for a given dataset labeled with genre information, when the pattern indexed by 10 appears in a recording, we can say that recording belongs to the label "classical" with 100% of confidence. Also, if the patterns 23 and 4 happens in the same recording, it belongs to the label "classical" with 94% of confidence. The patterns correspond to quantized subsequences of features – Mel-Frequency Cepstrum Coefficients (MFCC), in this case – and can be assessed visually or by listening to the music excerpt that generated it.

In this paper, we introduce algorithms to represent association rules in a graph, aiming to provide a visual tool to understand the relations between the features that compose it. Then, we apply our method on different datasets, described by varied classes of labels, to demonstrate how to use this representation to understand the relations between features and labels, as well as which patterns link two different labels.

* The opinions expressed in this article are those of the authors and do not necessarily reflect the official policy or position of the Itaú-Unibanco.

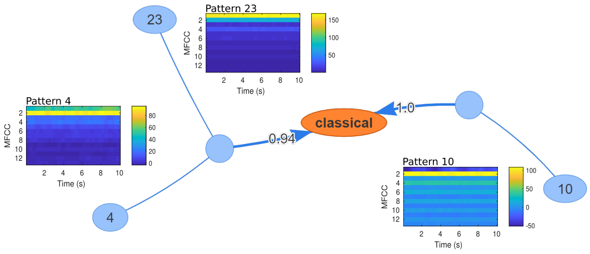

Figure 1: Example of association rule network

The remainder of this paper is organized as following. Section 2 introduces the main concepts of association rules and association rules networks, accompanied by related work. The method presented in this work is presented in Section 3. Section 4 presents our experimental evaluation. Finally, Section 5 concludes this work.

2. BACKGROUND AND RELATED WORK

The association rules were first proposed by Agrawal et. al. [1]. The goal of the proposed approach was identifying, in supermarket buying transactions, what were the items that customers used to buy together. This analysis was made aiming to help the supermarket owners to organize their stock in order to raise the sales of some specific items. To understand how association rules discovery works, we first define some concepts related to it.

Definition 1 *Given a set of items* I*, a set of transaction* T *consisting of subsets of* I*, an association rule is the relation* A $\rightarrow$ B*, where* A *and* B *are subsets of* I *and* A $\cap$ B = $\emptyset$.

A is called *antecedent* (or Left-Hand side - LHS) and *B* is called *consequent* (or Right-Hand side - RHS). The association rule can be read as: "given that *A* happened, *B* also happens in $c\%$ of the cases", where $c\%$ is the association rule confidence. Support ($s\%$) is another important measure in the association rule, that describes the percentage of transactions in which all the items of the rule appear.

Definition 2 *The support* σ(A) *of a subset* A $\subset$ I *is defined by the percentage of transactions that contain all the items presented in* A.

Definition 3 *The confidence of a rule* A $\rightarrow$ B *is given by the percentage of transactions that contain all the items in* A *that also contain all the items in* B*. The confidence is calculated by* $\frac{\sigma(A \cup B)}{\sigma(A)}$.

Also, the Lift is a widely used measure to assess the association rule quality. It evaluates if the items on the LHS are positively or negatively dependent with the items on RHS, or if these sets are independent.

Definition 4 *The lift value of a rule* A $\rightarrow$ B *is given by the probability of* A *and* B *happen together divided by the probability of* A *times the probability of* B*, calculated by* $\frac{\sigma(A \cup B)}{\sigma(A)\sigma(B)}$

The Association Rule Network (ARN) was proposed by Chawla et. al. [5] and extended by Pandey et. al. [10] and by Chawla [4]. The ARN models all the association rules that are directly or indirectly correlated to an specific item (called objective item) in a directed acyclic graph (DAG), pruning all the other rules that are not interesting in the objective item context. According to Pandey et. al. [10], the ARN modeling is capable of pruning the rules into a specific context, defined by the selected objective item.

According to Thulasiraman et. al. [14], a graph G = (V, E) consists of two sets: a finite set of vertices V and a finite set of edges E. Each vertex represents an object in the graph and an edge represents a link between two vertices. Also, it is possible to define the graph G = (V, E, W), consisting of three sets: the V and E sets remain the same, while the W set represents the weight of the edges in the graph G. In a graph that does not have weights, the W may have 1 where the connection exists and 0 where it does not exists. If the edges are ordered, i.e., the edges are identified as "from" vertex and "to" vertex, then it is said that the graph is directed because its edges contain a direction. A Directed Acyclic Graph (DAG), is a particular type of graph that contains no cycles.

Definition 5 *We say that a directed graph contains cycles if given a graph G containing N vertices V, the graph has a path that goes from* v_x *to* v_y *and there is also a path from* v_y *to* v_x.

An example of an ARN with objective item "G" is presented on Figure 2. In this example, the following rules were modeled: $A \rightarrow D$, $B \rightarrow D$, $B \rightarrow C$, C *&* $D \rightarrow E$ and E *&* $F \rightarrow G$. All the other extracted rules were pruned because they were not interesting in the context of the *G* item exploration.

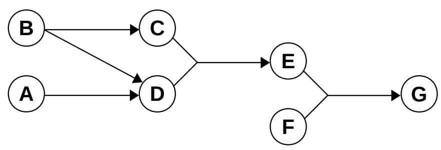

Figure 2: Example of ARN with objective item = *G*, adapted from Pandey et. al. [10].

The final ARN is a directed acyclic graph that flows to the objective item, i. e., all elements on the graph have directed connections that leads to an objective item. This graph models the association rules that better explains the occurrence of the selected objective item. The modeling can be used to build a hypothesis, based on the correlation among the items in the database and the objective item that the user wants to understand.

The ARN algorithm can be described in 3 steps, described as follows.

Step A Given a database D, a minimum support, and a minimum confidence value, we extract the association rules using an algorithm, like apriori [1]. The RHS must have size 1.

Step B Considering all the items in the association rules' RHS, the user selects one item to represent the objective item.

Step C Models all the rules that have the objective item in the RHS (considered level 0) or already modeled on other levels. The modeling must fulfill the 2 restrictions: 1 - The LHS of the rule is not present in the network or 2 - the LHS level is equal the RHS level + 1.

3. EXTRACTING AND ASSOCIATING PATTERNS

The proposed method relies on two main steps, which we describe in details in this section. The first one extracts frame-level features from the audio and quantize them to create a limited dictionary of patterns. With this procedure, we transform the recordings in our database in a transaction-like representation, where each recording is represented by the patterns presented in it. This representation is used in the second step of our method.

The second one extracts and selects the best rules to describe all the labels on the dataset. We extract the association rules, prune the rules which do not present interesting knowledge on the labels context and build a DAG, explaining how the patterns correlates to the labels.

3.1 Extracting and Quantizing Subsequence Patterns

As aforementioned, the first step of our method relies on associating each recording to one or more temporal patterns in a bag-of-patterns representation. For this, we split this phase into different intermediate steps to create a dictionary and, then, associate each subsequence of features from each recording to a codeword in this dictionary.

Initially, we extract frame-level Mel-Frequency Ceptrum Coefficients (MFCC) and Constant-Q chromagram from the raw audio. For this, we used the LibROSA package for music and audio analysis [9]. Since our main purpose is not comparing different parameter settings of each feature extraction procedure, we applied the default parameters defined by the tool. We chose these features since they represent distinct characteristics of music. Specifically, the MFCC and chromagram are intrinsic to timbre and pitch information, respectively.

To associate each feature vector to a pattern, we first need to create a dictionary. For this, we applied the simple k-Means clustering algorithm on subsequences of the feature vectors. The centroid of each cluster defines one codeword, i.e., the prototype of a temporal pattern. For the sake of memory and time efficiency, we only used one-third of the subsequences to estimate the centroids.

Once the codewords are defined, we associate all feature subsequences of each song with codewords, according to their proximity. In other words, for each recording, we find the nearest centroid of each subsequence of features and annotate it. At the end of this step, each recording is described by the set of codewords that appear in its subsequences. In the case of repetition, we remove these recurrences.

Although this step relies on defining the number of codewords, we leave details regarding this to Section 4.

3.2 *Extended* Association Rule Network

The *Extended* Association Rule Network (ExARN) aims to aid the user to understand the data and build a hypothesis from that data. The objective is to model the association rules in a graph, explaining the correlation among the attributes in the database according to a set of attributes selected by the user. For instance, consider the contact lens database [1] , which is aimed to automatically prescribes contact lenses to patients. In this case, the user may be interested not in the classification, but in understanding which are the patient's characteristics that define which kind of lens will be prescribed.

The ExARN is conceptually different from associative classification algorithms and decision trees, which build the model only based on the classes, ignoring all other correlations present on the database. The ExARN explores a set of previously extracted association rules and, then, searches for the best rules to describe the set of attributes defined by the user. Also, it has some interesting properties: i) it is built on a DAG, which means that there are no cycles on the network, ii) it is built on levels, every rule has the LHS on level x and the RHS on level $x + 1$, for example, an objective attribute will mandatorily be on the RHS and on level 0, all the attributes on LHS that contains that attribute on RHS will be on level 1 and so on.

The ExARN is built in three steps. The first step consists of the association rule mining phase. The only restriction added to this step, if compared to a conventional association rule mining, is that the rules must an RHS with size 1. This restriction was added, so each rule explains only one attribute, reducing the complexity for the user to explore the result.

The second step is the objective attribute definition. This step will guide the entire exploration, as it will define the objective attributes which the network will be built from. The user must select the attributes that will be explored. This selection must be done considering also the possibility that these attributes have a common cause to be explored or refuted.

The last step consists of the *Ex*ARN construction. This step is responsible for getting all the rules that are directly or indirectly related to the objective attributes and model them following the *Ex*ARN restrictions. The *Ex*ARN building is done recursively. First, all the attributes selected as objective attributes are modeled in the graph on the level 0. Then, all the rules that the LHS' attributes are not in the graph and have the RHS' attributes on level 0 are modeled on the network. The same process is done to all the attributes on level 1, then to attributes on level 2 and so on. Until there are no more rules to be modeled. The *Ex*ARN can be defined as follows:

Definition 6 *Given a set of association rule R, containing rules with RHS of size 1, and a set of objective attributes*

[1] `https://archive.ics.uci.edu/ml/datasets/lenses`

Z with size ≥ 2, the Ex*ARN is a DAG that models all the rules related to the items on Z, such as:*

1. Each vertex models a rule r $\subset$ R.

2. From any point of the network, it is always possible to reach at least 1 vertex representing an attribute from Z.

3. Given a vertex v $\subset$ *Ex*ARN, such as v $\not\subset$ Z. There is no path from any item on Z to v.

The *Ex*ARN presents a wider exploration if compared to the presented ARN because it allows the exploration of 2 or more objective items at once. That way, the user might discover which patterns are interesting in the context of a single objective item, also discovering which patterns are interesting for a set of objective items.

4. EXPERIMENTAL EVALUATION

In this section, we describe the experimental evaluation performed to assess the ExARN in different scenarios of music data. We note that we made a supplementary website [2] where we make available source codes and detailed results, as well interactive visualizations of the networks presented in this section and some audio excerpts to exemplify some of the mentioned patterns.

4.1 Rule discovery and association setup

After extracting the subsequence patterns, the database pattern extraction begins. The sequence described here was applied to all the databases.

First, the association rules were extracted. We mined the association rules using the arules package in R [3]. This step needs the definition of some parameters. The support value, which is a threshold of minimum occurrence was set to 1%. This value was chosen because the databases were divided into more than 10 different labels, so each subpattern will have a maximum occurrence of $\frac{1}{numLabels}$ on each label. Defining the minimum support to 1% will remove only the subpatterns that rarely happens. The other parameter is called confidence, which can be defined in terms of posterior probability as: $Conf(A \rightarrow B) = P(B|A)$. We defined the minimum confidence on 25%, which means that the B must happen in, at least, one-quarter of the occurrence of A.

To make sure that the association rules are positively dependent, we applied a filter using the lift measure. The threshold was defined at 2, as rules with lift value ≥ 1 are considered to have a positive dependency. We selected the value 2 instead of 1 as this value discards the rules that are on the edge of the measure. Then, we applied the ExARN algorithm over the association rules, considering all the labels as the objective items

[2] https://sites.google.com/view/music-exarn

[3] available at https://cran.r-project.org/web/packages/arules/index.html

4.2 Datasets

The datasets used in our experimental evaluation aim to provide us a diversity of characteristics and labels. For this reason, we used diversified datasets and for one of them, we used different labels for the same bag-of-patterns.

One of the most common labels in MIR datasets is the genre. We evaluated our method in this context using the GTZAN dataset [15]. This database is composed of 1000 thirty-second tracks, perfectly balanced in ten genres.

Another way to categorize music data is according to the artist who recorded it. We also evaluated our method in this scenario, using the Artist20 dataset [7]. This dataset contains 1413 songs performed, as its name suggests, by 20 artists mostly of pop or rock music. The number of recordings is not balanced among the artists.

Finally, we assessed the FMA dataset [6]. Moreover, we took the fact that many of the recordings in these databases are associated with "social" features provided by Echonest [4] to evaluate our method on varied labels for the same data. Specifically, we applied our method targeting seven distinct labels: acousticness, danceability, energy, instrumentalness, liveness, speechness, and valence. In order to transform these continuous features in class-like values, we discretized the features in five equally spaced intervals, representing low, mid-low, mid, mid-high, and high levels of each characteristic. As we used the default small portion of this data and only kept information from the recordings associated with Echonest features, we ended with 1023 tracks.

4.3 On the Impacts of the Codebook's Size

The quantizing phase of our method has one parameter that affects the results of our method. The number of clusters to create the dictionary, i.e., the number of codewords, have a direct impact on the confidence. Particularly, the higher the number of codewords, the lower the confidence of the rules. Conversely, the lower the number of clusters, the higher is the confidence.

As we experimented with 25, 50, and 100 codewords in each dataset, we stick our analysis on the lower value. However, we notice that a high number of codewords may be more appropriate for datasets with a high number of labels. Otherwise, the intersection between the labels would be too high to find meaningful rules. In this paper, the higher number of assessed labels is 20.

There is another characteristic of using fewer codewords regarding the interpretability of the results. As the codewords are centroids, if we use too few clusters the codewords will look "blurry" or few informative. However, we noticed that it does not hamper the rule discovery and the music excerpts that were associated with each pattern can listen to a better understanding of what that pattern represents. Also, once the ExARN is computed, we can break a pattern A in more parts, B and C for instance, in a procedure similar to the Bisect k-Means approach [13]. With this operation, we turn the patterns more specific.

[4] http://the.echonest.com/

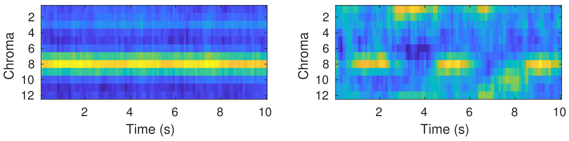

Figure 3: Examples of patterns in different resolutions, given by the number of codewords in the dictionary: 25 (*left*) and 100 (*right*)

Then, instead of reading the association regarding A, one may read it regarding "B or C." Figure 3 illustrates two chroma patterns obtained from different number of codewords.

4.4 Results and Discussion

In this section, we present some of the results obtained by our method. For simplicity, we split it into sections regarding each analyzed dataset and the target labels of each of them. Specifically, the results on GTZAN, Artist20, and FMA are presented in the sections regarding genre, artist, and the "social features" from Echonest.

We acknowledge that interpreting the patterns and, as consequence, the meaning of some rules, only using textual and static graphical elements is a difficult matter. For this reason, we make available on our website interactive visualizations of ARNs obtained in our experiments, as well as some music excerpts that are representative of relevant patterns.

We note that association rules discovery is an unsupervised task and, therefore, there is no quantitative evaluation measure to assess the quality of these rules. The only way to objectively evaluate the value of the learned rules would be use it as an intermediate step of an algorithm to perform other task, such as classification or recommendation systems. We leave this as an intention for future work.

4.4.1 Genre

Using MFCC, we found a few interesting rules that associate patterns with some of the target labels. One example is the one illustrated by Figure 1. We also found similar associations to other genres. Specifically, for metal, reggae, and jazz. The latter two, however, with lower confidences (around 33%).

The most interesting relations in this dataset come from the patterns shared between distinct labels. Figure 4 presents the entire network for this dataset when associating patterns representing 10 seconds of audio.

Some of the patterns are associated with several genres in the presented network. These patterns are not suitable for differing the characteristics of each genre. However, we commonly see music elements that are used in songs belonging to different genres. So, this kind of multiple relations was expected. For instance, we observed a pattern associated with the genres disco, pop, and hip-hop.

On the other hand, we found patterns that link pairs of genres. These patterns directly associate two genres that

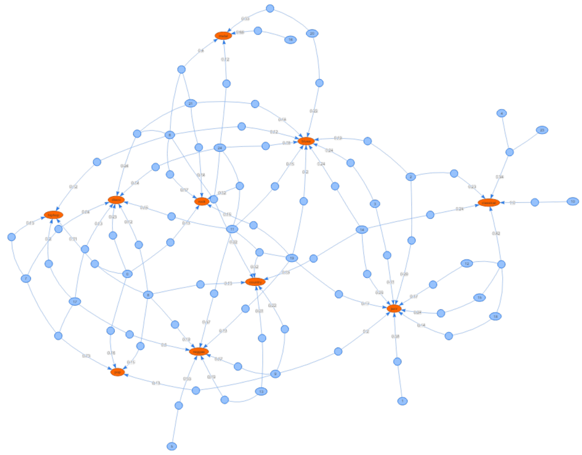

Figure 4: ExARN obtained by associating MFCC patterns from 10 seconds of audio in the GTZAN dataset

have somehow similar timbral information. It may help explain the influence between genres or the mutual influences of each pair. One example of two genres linked by this kind of association is the pair metal and blues. Another interesting relation regards the genres classical and jazz. They have two patterns that are common for both. However, they usually happen together (i.e. in the same recording) in classical pieces but separately in jazz songs. We noticed that we did not achieved interesting associations when using chroma features in this case.

4.4.2 Artist

Is there any link between Metallica and Roxette regarding tonal patterns in their songs? The answer is "yes, there is Tori Amos." Using subsequences of chroma vectors representing 10 seconds in the Artists20 dataset, we found that these three artist have sets of four tonal patterns each that are confidentially linked to each of them. Moreover, Tori Amos shares one of its patterns with Metallica and another one with Roxette. Figure 5 illustrates these relations.

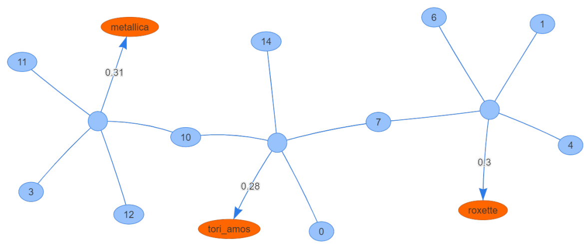

Figure 5: Subset of the rules obtained from chroma patterns in the Artist20 dataset

This kind of relation is commonly seen when using MFCC as the input features in this dataset. For instance, when applying the ExARN algorithm on five seconds excerpts, we found rules with (at least) the minimum support for seven artists. When using ten seconds excerpts, we found rules for ten artists. In all of these cases, there is at least one timbral pattern for each artist which links it to another one.

We notice that these links are relevant since they are not trivial. In other words, if a timbral or chromatic pattern is present in many songs of several artists, the rules containing it would have a very low support. Therefore, these links show how two (or more artists) are musically related each other by patterns that are not so commonly used.

4.5 Echonest Labels

We evaluated the ExARN on seven different labels from Echonest. We found relevant rules in all of them but the *speechiness*. For the other labels, the association rules network demonstrated regularities in their behavior. For instance, when we assess the rules associated to a single label, usually we cannot find association with minimum support for the intermediate values. This may happen because the middle labels are fuzzy. In other words, the assessed patterns can describe solely the high and low characteristics at a minimum support. Figure 6 illustrates this fact regarding the acousticness.

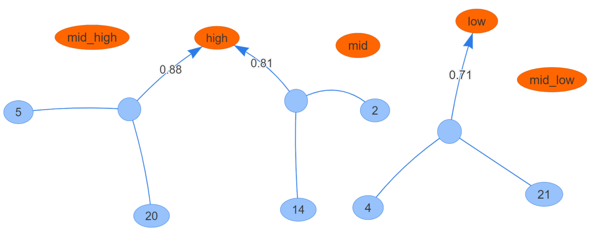

Figure 6: Association rules from MFCC that are not shared by different intervals of acousticness

When we analyze the rules that associate different labels, three main behaviors appear. The first one is not finding patterns which relate different levels of these characteristics. Figure 7 illustrates the second, and most common, behavior. In this case, the extremes are separate into distinct components, i.e., the high and mid-high values are linked by some patterns, similarly to what happens between low and mid-low values.

Finally, in some cases, the labels representing extreme values are directly linked by one or more patterns while some patterns play the rule of "bridges" between these extremes. Figure 8 illustrates this scenario.

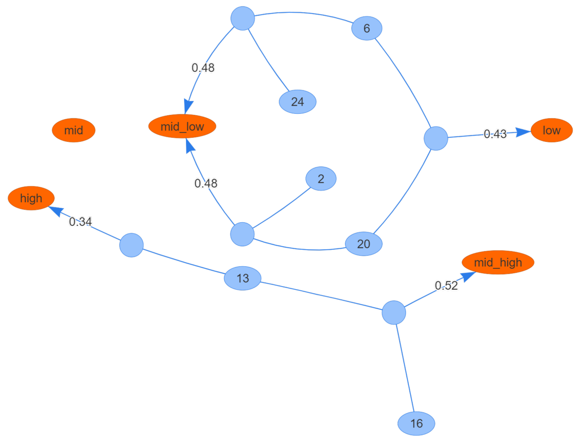

Figure 7: Association rules from MFCC shared by different intervals of energy

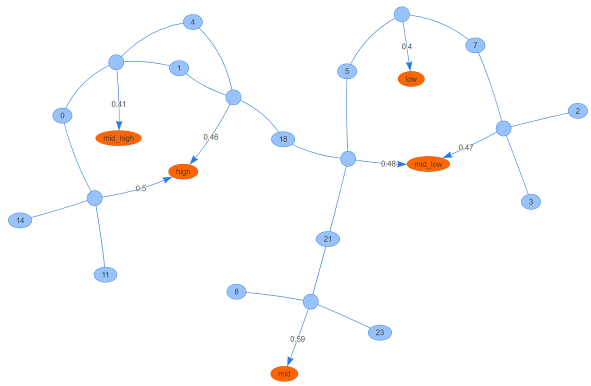

Figure 8: Association rules from chroma features shared by different intervals of energy

5. CONCLUDING REMARKS

In this paper we presented the use of extended association rules networks for exploring the correlation between temporal patterns and labels of music in different scenarios.

To evaluate the meaning of the discovered rules, we presented some reasoning to verify the quality of these rules as a qualitative approach. One example is evaluating the existence of links between labels we consider similar to each other. In other cases, our rules may explicit some relations that are not obvious. In both cases, studying the patterns that composes such relations can be useful to understand music data in several aspects. For instance, in some cases, we could find interesting relations using chroma vectors in scenarios where these features are not usually considered (e.g. to describe valence and energy).

As future work, we intend to improve the quantization step so we reduce the impact of the codebook generation and we can ignore several patterns, considering them irrelevant. For this, we may evaluate the use of some density-based clustering strategy [3,11]. Also, we will evaluate the use of ExARN as an intermediate step to improve recommendation systems. Finally, we intent to evaluate if this kind of association rules network can improve the interpretability of music-related learned features.

6. ACKNOWLEDGEMENT

This work was funded by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) and São Paulo Research Foundation (FAPESP) by grants #2013/26151-5, #2016/17078-0, and #2018/11755-6.

7. REFERENCES

[1] Rakesh Agrawal, Tomasz Imielinski, and Arun Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD International Conference on Management of Data*, pages 207–216, 1993.

[2] James Bergstra, Norman Casagrande, Dumitru Erhan, Douglas Eck, and Balázs Kégl. Aggregate features and AdaBoost for music classification. *Machine learning*, 65(2-3):473–484, 2006.

[3] Ricardo J. G. B. Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 160–172, 2013.

[4] Sanjay Chawla. Feature selection, association rules network and theory building. In *International Workshop on Feature Selection in Data Mining*, pages 14–21, 2010.

[5] Sanjay Chawla, Bavani Arunasalam, and Joseph Davis. Mining open source software (OSS) data using association rules network. In *Advances in Knowledge Discovery and Data Mining*, pages 461–466, 2003.

[6] Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. Fma: A dataset for music analysis. In *International Society for Music Information Retrieval Conference*, 2017.

[7] Daniel P. W. Ellis. Classifying music audio with timbral and chroma features. In *International Society for Music Information Retrieval Conference*, pages 339–340, 2007.

[8] Philippe Hamel and Douglas Eck. Learning features from music audio with deep belief networks. In *International Society for Music Information Retrieval Conference*, pages 339–344. Utrecht, The Netherlands, 2010.

[9] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Python in Science Conference*, pages 18–25, 2015.

[10] Gaurav Pandey, Sanjay Chawla, Simon Poon, Bavani Arunasalam, and Joseph G. Davis. Association rules network: Definition and applications. *Statistical Analysis and Data Mining*, 1(4):260–279, 2009.

[11] Alex Rodriguez and Alessandro Laio. Clustering by fast search and find of density peaks. *Science*, 344(6191):1492–1496, 2014.

[12] Diego F Silva, Chin-Chia M Yeh, Gustavo Enrique de Almeida Prado Alves Batista, and Eamonn Keogh. Simple: Assessing music similarity using subsequences joins. In *International Society for Music Information Retrieval Conference*, pages 23–29, 2016.

[13] Michael Steinbach, George Karypis, Vipin Kumar, et al. A comparison of document clustering techniques. In *ACM SIGKDD Workshop on Text Mining*, pages 525–526, 2000.

[14] K. Thulasiraman and M. N. S. Swamy. *Graphs: Theory and Algorithms*. 1992.

[15] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.