Technical exercise Deezer
Relations and similarities between “musical tags”

Author: Gabriel DELGADO (gabriel.delgado@alumni.polytechnique.org)

Dataset description

This dataset contains social networking, tagging, and music artist listening information from a set of 2K users from Last.fm online music system http://www.last.fm. The dataset is released in the framework of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011) http://ir.ii.uam.es/hetrec2011 at the 5th ACM Conference on Recommender Systems (RecSys 2011) http://recsys.acm.org/2011. This dataset was built by Ignacio Fernández-Tobías with the collaboration of Iván Cantador and Alejandro Bellogín, members of the Information Retrieval group at Universidad Autonoma de Madrid (http://ir.ii.uam.es).

The dataset represents the information of 1892 users, 17632 artists, 12717 bi-directional user friend relations, 92834 user-listened artist relations, 11946 tags and 186479 tag assignments. For more details consult the file “readme.txt”.

Objective and hypothesis

The objective of this exercise will be to describe the relations and similarities between different musical tags included in the aforementioned dataset. Since we desire to identify the inherent features within the data without any numerical or categorical label, this exercise corresponds to a typical non-supervised machine learning problem. In our case we will try to separate the musical tags into groups (clustering) according to a certain defined affinity which hides the idea of musical preferences and styles (topics). The three following underlying hypothesis will guide the proposed data treatment:

1. Every user possesses a predominant musical preference which can be described by a certain number of musical tags;
2. Every artist possesses a predominant musical style which can be described by a certain number of musical tags;
3. The musical preferences and the musical styles can be described through similar musical tags.

The ensuing analysis will be performed using Python as programming language.

Methodology

For the present study we apply the following steps:

1. **Data cleaning and munging**

   The data is loaded using the Pandas library. No missing values are found in the dataset. For a matter of limited time, we focus our attention on musical tags that are used more than once (we reduce thus the initial 11946 tags to 1090 tags) and the top three artists for each user according to their listening count. From the data we also remark that not all the artists listened by a user have being tagged nor all the artists tagged by some user are necessarily listened by this person.

2. Definition of a measure of dissimilarity or distance between musical tags

Let “A_{ij}” be the number of artists that have simultaneously been tagged with the tags “tag_i” and “tag_j”. This information can be retrieved from the file “user_taggedartists.dat” considering the set of tags associated to an artist. Also let “U_{ij}” be the number of users for whom their 3 most listened artists have been tagged with the tags “tag_i” and “tag_j” (not necessarily the same artists). This information can be retrieved from the files “user_taggedartists.dat” and “user_artists.dat”. Then according to the hypothesis mentioned in the previous section (“Objective and hypothesis”), we propose two distance (or dissimilarity) measures between two different musical tags:

- M_1(tag_i,tag_j) = $\exp(-\alpha A_{ij})$, $\alpha>0$, $i\neq j$
- M_2(tag_i,tag_j) = $\exp(-\alpha U_{ij})$, $\alpha>0$, $i\neq j$.

The first measure M_1 is the simplest one and it is supposed to account for the hypothesis “Every artist possesses a predominant musical style which can be described by a certain number of musical tags”. Indeed, given two different musical tags, if more artists have been tagged with these tags then we can suppose that these tags are close and represent the same musical style. The second distance measure M_2 considers the hypothesis “Every user possesses a predominant musical preference which can be described by a certain number of musical tags”. In fact, given two different musical tags, if more users prefer artists that have been tagged with these tags, then we can suppose that these tags represent the same musical preference.

In both cases we use the negative exponential function in order to reflect the fact that two tags are closer (i.e. the distance approaches to zero) if the values of A_{ij} and U_{ij} are larger. It also naturally handles the case when A_{ij} or U_{ij} are zero.

We expect that the distance measures M_1 and M_2 will lead us to similar clusters according to the announced hypothesis “The musical preferences and the musical styles can be described by similar musical tags.”

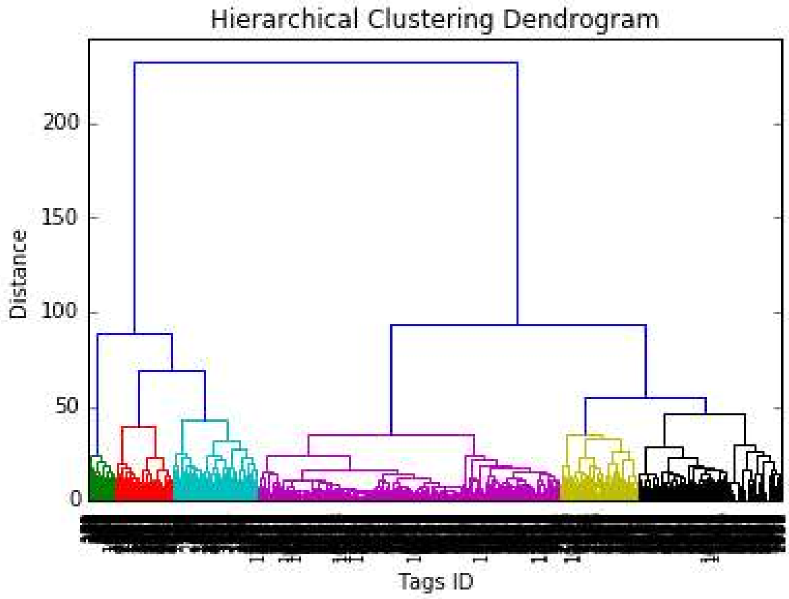
3. Musical tags clustering

We perform a hierarchical/agglomerative clustering with the SciPy library. This approach has the double advantage with respect to other clustering techniques (such as k-means or k-medoids), of being applicable to non-numerical data for which we cannot define the mean within the clusters and also allow the user to define a posteriori the number of desired clusters. Also we use the Ward linkage distance for computing the cluster distance. Even though for a proper utilization of the Ward distance the supplied distance matrix should be Euclidean (i.e. if two tags are closely related to a third, then they must be related), we admit that the dataset satisfies more or less this condition for a matter of limited analysis time.

Results

1. **Dissimilarity measure M_1 with α=1**

   Taking maximum depth “d=50”, the generated dendrogram identifies 6 clusters:



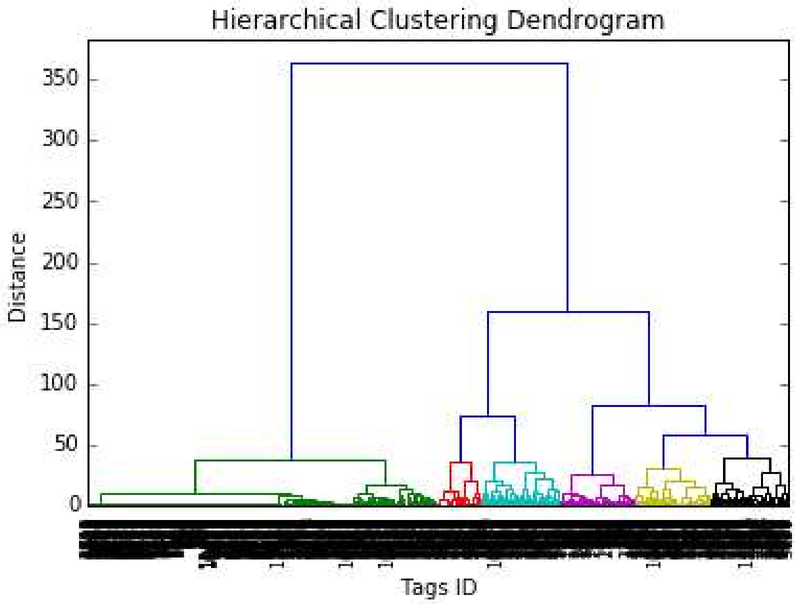
The choice of “d” intends to construct clusters with similar sizes. Here below we list a few musical tags within each cluster:

C1 (green) = [chillout, electronic, pop, dance, alternative, cool, favorites …]
C2 (red) = [hiphop, rap, pop dance, party, britney spears, lady gaga …]
C3 (cyan) = [ambient, new wave, new age, synth pop, 80’s, psychedelic …]
C4 (magenta) = [agrotech, cyberpunk, ska revival, jazz fussion, Eminem, back sabath, …]
C5 (yellow) = [alternative metal, black metal, death metal, industrial, noise, german …]
C6 (black) = [world music, ethnic, progressive trance, fusion, italian, indie folk …]

Although the choice of the above listed tags from the total set of tags is probably biased by the author according to expected musical styles (the author was incapable of leaving Black Sabbath and the Backstreet Boys in the same cluster!), a certain musical coherence naturally arises within each cluster, namely C1 (electronic-pop), C2 (hiphop-rap), C3 (new age-psychedelic), C4 (cyberpunk-ska), C5 (metal-hard rock) and C6 (ethnic-international).

2. **Dissimilarity measure M_2 with α=0.1**

   Taking maximum depth “d=50”, the generated dendrogram identifies 6 clusters:



Here below we list a few musical tags within each cluster:

C1 (green) = [chanson française, metallica, mika, the killers …]
C2 (red) = [reggae, classic, celine dion, enrique iglesias …]
C3 (cyan) = [black metal, doom metal, hardcore, ethnic, green day …]
C4 (magenta) = [alternative rock, indie, new wave, electronic, love, romantic …]
C5 (yellow) = [synth pop, pop dance, kylie minogue, lady gaga, queen, female voices …]
C6 (black) = [punk rock, gothic metal, heavy metal, u2, Marilyn manson …]

Compared to the results obtained with the first dissimilarity measure M_1, this results barely separate the tags into known musical styles representing each cluster. This difficulty may be caused by the following reasons: the choice of α=0.1 (instead of 1) motivated by the low values of $\mathbf{M_2}$ for some pairs of tags, the choice of 3 top artists for each user (instead of more), the wrong supposition that the spawned distance matrix was Euclidean in order to use the Ward linkage distance, etc.

Next steps

Here below we list some recommendations and clues to pursue the data exploration in order to better understand the relations between musical tags:

- **Apply a multidimensional scaling (MDS) analysis**: Using for instance Scikit-learn, project the members of each one of the clusters previously identified on the first and second MDS coordinates in order to get a visual idea of the "spatial" proximity between them.
- **For a given dissimilarity metric, study the evolution of the clusters in time:** For each different year, perform the same analysis and check the clusters stability. Differences among clusters may be explained following the emergence of new musical styles (new artists or new user preferences).

- **Add to the users metric M_2 also the notion of friends:** Assuming that friends have similar musical preferences, modify the definition of U_{ij} in order to also consider the preferences of friends (“user_friends.dat”).
- **Consider the list of musical tags neglected in this study:** Once the right metric defined, consider in your analysis the musical tags mentioned only once by a user.
- **Construct a recommendation algorithm:** Once the musical tags clusters are defined and validated, build a recommendation algorithm to propose new artist to a user from his historic preferences in terms of musical tags, listened music and friend’s preferences.