Introduction to Data Mining, CSE 5243

Final Project

EMI Music Data Science - Predicting User-Track Rating Based on User Profile and Artist Word Association

Shiuli Das (das.190), Debanjan Nandi (nandi.20)

1. OBJECTIVE

How do people describe the music they have just heard, and connect to it? The EMI Dataset performs extensive market research about artists by interviewing thousands of people around the world. This research has produced the EMI One Million Interview Dataset Design. “The EMI Million Interview Dataset is the richest and largest music dataset ever, a massive, unique, rich, high-quality dataset that contains interests, attitudes, behaviors, familiarity, and appreciation of music as expressed by music fans around the world”.

The Data Science London Hackathon, held on the weekend of July 21st and 22nd, 2012, took a sample of the data from the United Kingdom that provides a granular mixture of profile, word-association, and rating data.

The goal of the hackathon, as well as ours, was to design an algorithm that combines (a) user demographics, (b) artist and track ratings, (c) answers to questions about their preference in music, and (d) words that they use to describe EMI artists to predict how much they like the tracks they have just heard.

2. DATASET DESCRIPTION

We were provided with five files:

a. Train/Test Data:

This csv file contains data that relate to how people rate EMI artists, during market research interviews, right after hearing a sample of the artist’s song. The 6 columns are:

- Artist: An anonymized identifier for the EMI artist
- Track: An anonymized identifier for the artist’s track.
- User. An anonymized identifier for the market research respondent, who will have just heard a sample from the track.
- Rating: A number between 0 and 100 which answers the question: *How much do you like or dislike the music?* (Train only, this is being predicted for the test set)
- Time: The time the market research was completed. It is anonymized research date indicating which month the research was conducted in. It can be used to understand which other artists/ tracks were researched in the same wave. It was noted that this data was not in a chronological order

b. Words:

This csv file contains data that shows how people describe the EMI artists whose music they have just heard.

- Artist: An anonymized identifier for the EMI artist
- User: An anonymized identifier for the market research respondent, who will have just heard one or more samples from the artist.
- HEARD_OF: An entry which answers the question: Have you heard of and/or heard music by this artist.
- OWN_ARTIST_MUSIC, which answers the question: Do you have this artist in your music collection.
- LIKE_ARTIST. A numerical entry which answers the question: To what extent do you like or dislike listening to this artist?
- Finally, a list of words. There are 82 different words, ranging from “Soulful” to “Cheesy” and “Aggressive”. After listening to tracks from an artist, each respondent had selected words they thought best described the artist from a given set. The values in each column were therefore 1, if the respondent thought that word described the artist, 0 if the respondent did not think the word described the artist, and empty if the word was not part of the current interview set.

c. UserKey and users:

The final csv file gives the data about the respondents themselves, including their attitude towards music. The columns include:

- User: The anonymized user identifier
- Gender: Male/Female
- Age: The respondent’s age, in years
- Working status: Whether they are working full-time/retired/etc.
- Region: The region of the United Kingdom where they live
- MUSIC: The respondent’s view on the importance of music in his/her life
- LIST_OWN: An estimate for the number of daily hours spent listening to music they own or have chosen
- LIST_BACK: An estimate for the number of daily hours spent listening to the background music/music they have not chosen.
- Music habit questions: Each of these asks the respondent to rate, on a scale of X-100, whether they agree with the following:
    - I enjoy actively searching for and discovering music that I have never heard before.

- I find it easy to find new music
- I am constantly interested in and looking for more music
- I would like to buy new music but I don't know what to buy
- I used to know where to find music
- I am not willing to pay for music
- I enjoy music primarily from going out to dance
- Music for me is all about nightlife and going out
- I am out of touch with new music
- My music collection is a source of pride
- Pop music is fun
- Pop music helps me to escape
- I want a multimedia experience at my fingertips wherever I go
- I love technology
- People often ask my advice on music - what to listen to
- I would be willing to pay for the opportunity to buy new music pre-release
- I find seeing a new artist/ band on TV a useful way of discovering new music
- I like to be at the cutting edge of new music
- I like to know about music before other people

3. SOFTWARE USED

We have used Python to preprocess our data and build our classification model. We used Weka for visualizing pre-and post-processed data and build classifiers.

4. PREPROCESSING

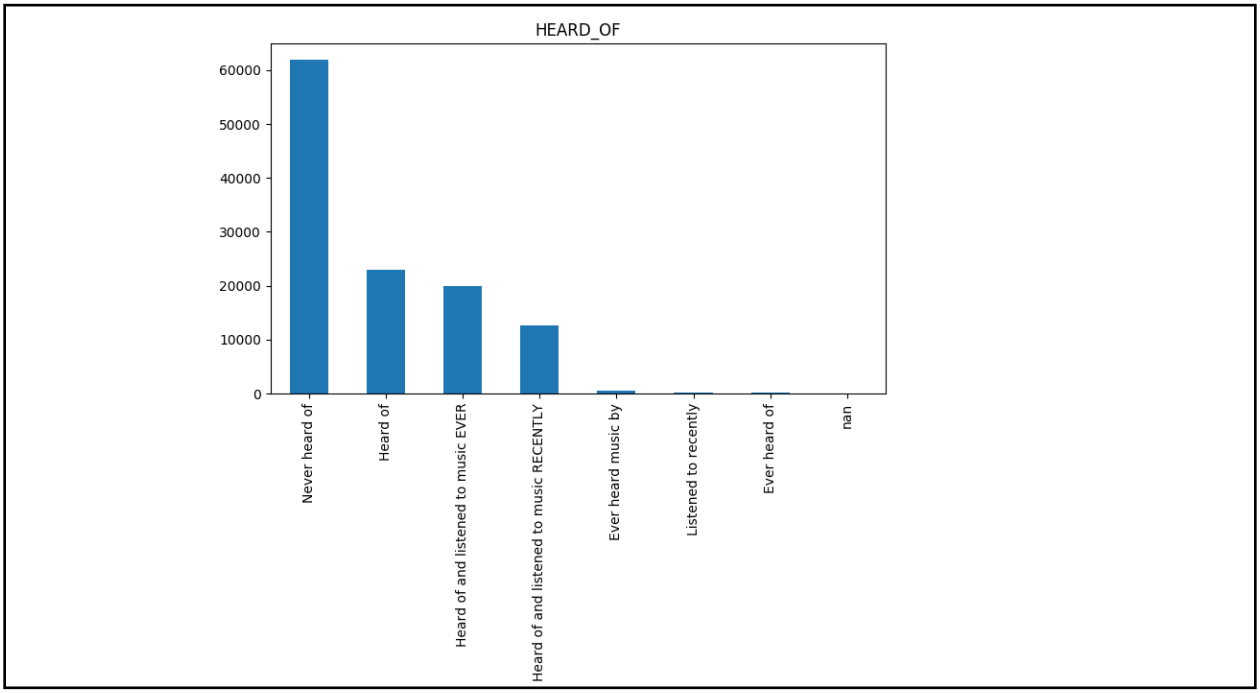
All the above datasets were processed, transformed, and modified to reduce the dimensionality of our feature vector per the following rules as follows

a. Words.csv:

i. Artist (nominal, 0 - 49): No processing operation
ii. User (nominal, 0 - 48644): No processing operation

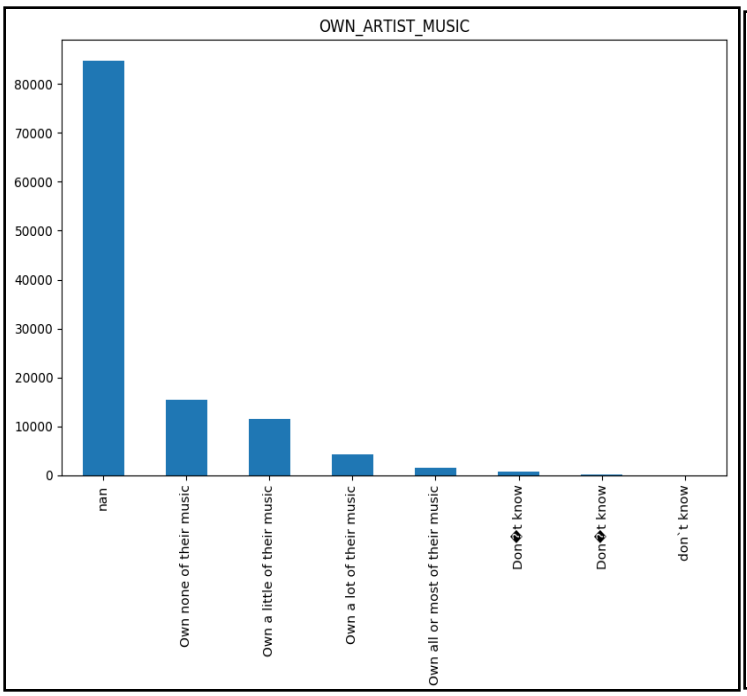
iii. HEARD_OF (nominal):

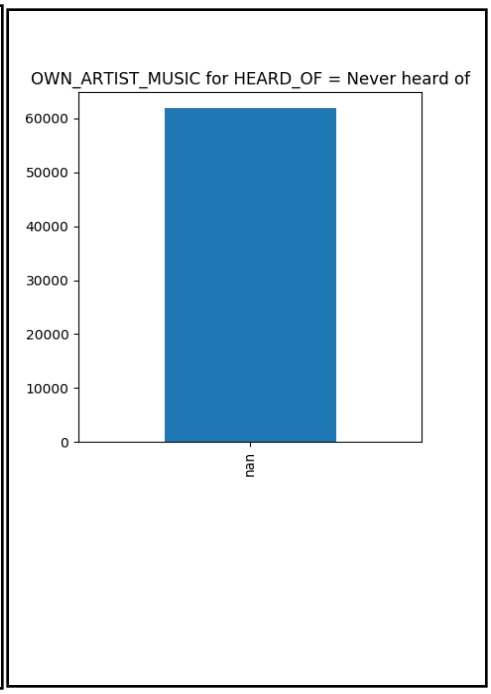
Attribute Value	New Value
'Heard of' 'Heard of and listened to music EVER' 'Heard of and listened to music RECENTLY' 'Ever heard music by' 'Listened to recently' 'Ever heard of	1
'Never heard of'	0



iv. OWN_ARTIST_MUSIC (nominal):

Attribute Value	New Value
'Own a lot of their music' 'Own all or most of their music'	2
'Own a little of their music' 'Dont know'	1
'Own none of their music'	0





v. If HEARD_OF == 0 (‘Never heard of’), then OWN_ARTIST_MUSIC = 0 ('Own none of their music')

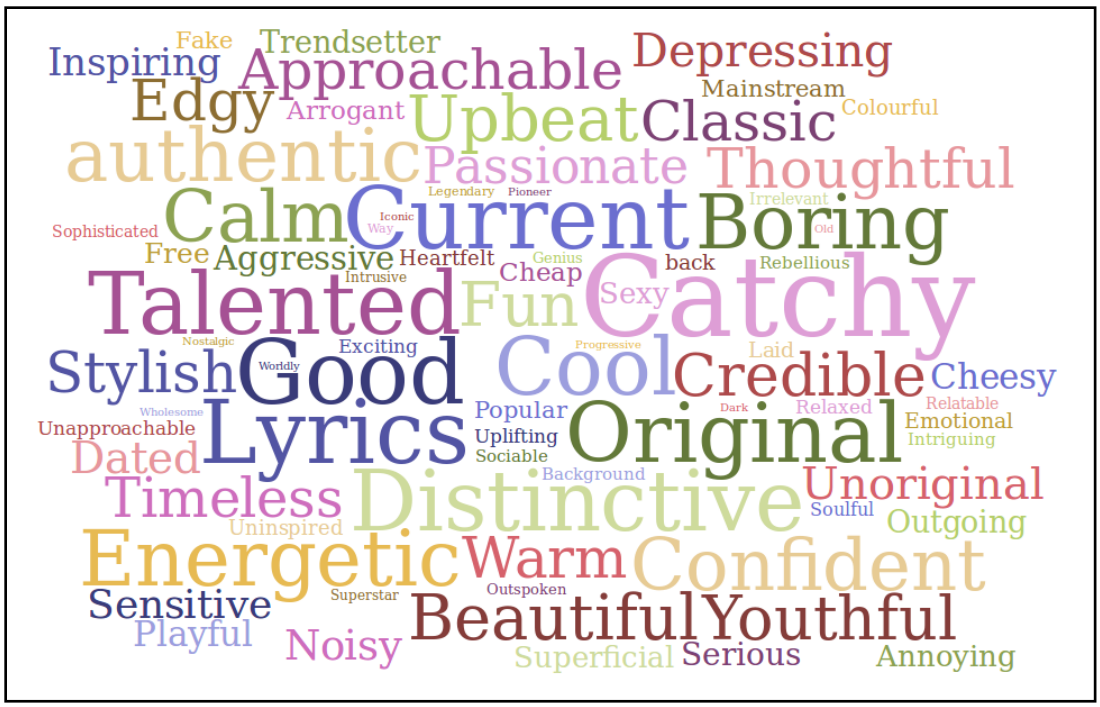
vi. LIKE_ARTIST (ordinal, 0 - 100): grouped into equal bins of size 10

vii. MISSING VALUES: All missing values were filled with 0

viii. Word List: The list of 82 different words were categorized into the positive, negative, and neutral sentiments it conveyed. We also divided them into categories depending upon what aspects of the song/ artist they described. The different categories are as follows:

Type	Positive	Neutral	Negative
Lyrics	Uplifting Intriguing Good lyrics Thoughtful Beautiful Inspiring Authentic Original Relatable Worldly Nostalgic	Sophisticated Serious Cheap Way out Sensitive Dark Depressing Emotional None of these Rebellious Intrusive Over	Uninspired Superficial Not authentic Cheesy Fake Unoriginal Irrelevant

Feel	Edgy Wholesome Colourful Heartfelt Fun Credible Cool Catchy Passionate Timeless Distinctive Upbeat Energetic Exciting Progressive Soulful Warm Relaxed Classic	Aggressive Laid back Free Old Youthful Current Calm Mainstream Background Dated Playful	Annoying Boring Noisy
Personality	Sociable Legendary Confident Stylish Pioneer Approachable Talented Genius Trendsetter Sexy Popular Superstar Iconic	Outspoken Outgoing	Unattractive Arrogant Unapproachable



WordCloud based on frequency count

The figure above visualizes what words the users chose to describe most of the artists based upon their usage. We observe that users mostly used positive and neutral words to describe the latter. A few negative words which have been used frequently used are Boring, and Noisy.

ix. ARTIST CLUSTERING:

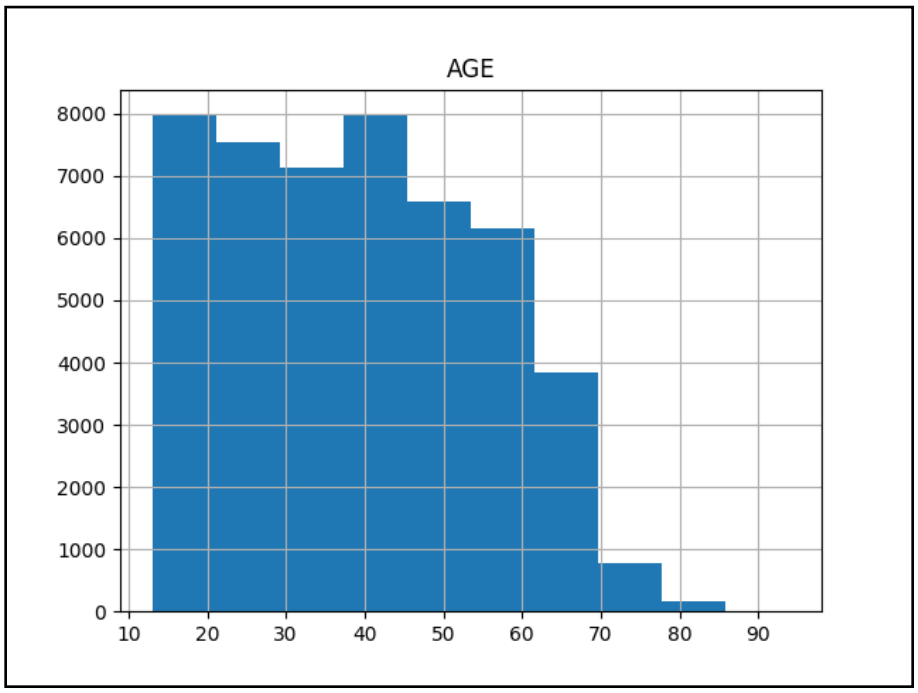
The different artists were clustered based on the type of words used by the interviewees (users) to describe their song. The 82-dimension word feature vector is reduced to 9 categories as described above. We used K-Means clustering to group the artists into 5 different clusters/ artist groups based on the average score each artist received corresponding to each word category.

b. Users.csv:

- RESPID (nominal, 0 -48644): anonymized user id
- GENDER (binary):

Attribute Value	New Value
Male	1
Female	0

- AGE (ordinal): divided into bins of size of 10. Bin values are as follows: 0-10, 11-20, so on, 91-100.

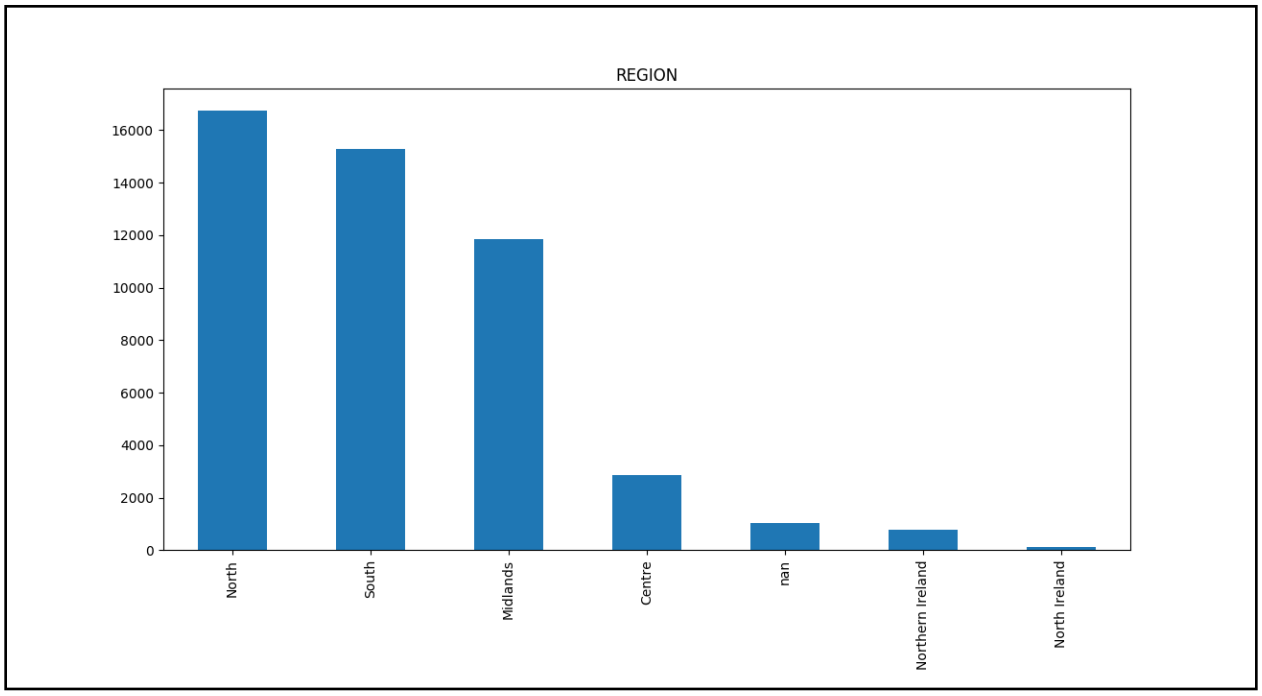


Missing Values:

This attribute had 467 missing values. The missing values were filled by randomly allocating bins strictly based on the probability distribution of the bins in the rest of the data set.

- REGION (nominal)

Attribute Value	New Value
North	North
South	South
Midlands, Centre	Midlands
Northern Ireland, North Ireland	Northern Ireland

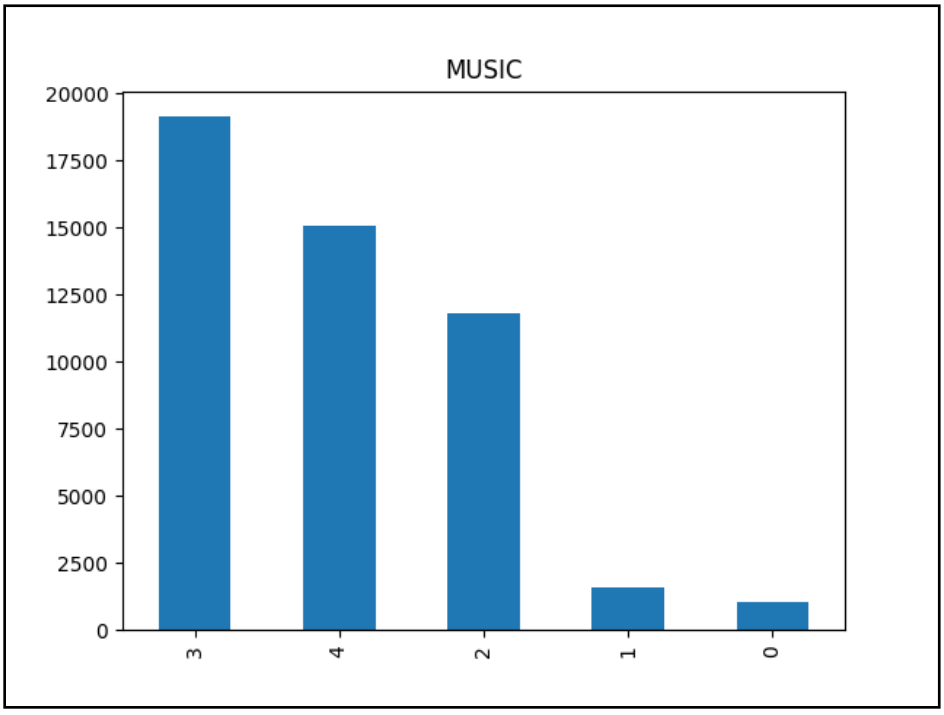


Missing Values:

This attribute had 1040 missing values. The missing values were filled by randomly allocating regions strictly based on the probability distribution of the regions in the rest of the data set.

- MUSIC (ordinal)

Attribute Value	New Value
'Music has no particular interest for me'	0
'Music is no longer as important as it used to be to me’	1
'I like music but it does not feature heavily in my life'	2
'Music is important to me but not necessarily more important than other hobbies or interests', 'Music is important to me but not necessarily more important	3
‘Music means a lot to me and is a passion of mine'	4

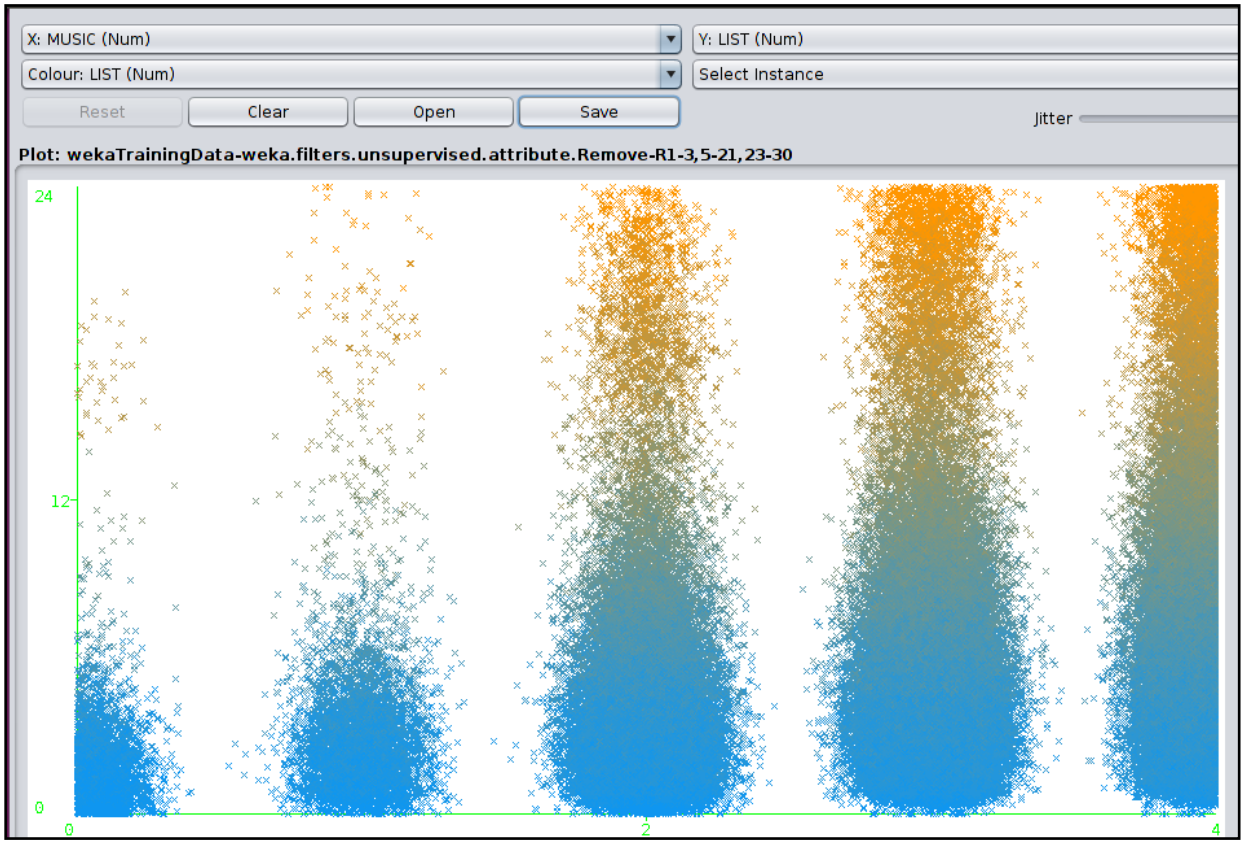


- LIST_OWN/LIST_BACK (ratio)

Attribute Value	New Value
const hour/s, const	const
More than 16 hours, 16+ hours	17
Less than 1 hour	1

We combined LIST_OWN and LIST_BACK together to form the attribute LIST which represented total time in hours spent on listening to music during a day. The maximum value for this attribute was clipped to 24 since we cannot have more than 24 hours in a day.

The attribute LIST had 8091 missing rows of data. We observed that there was a strong correlation between the number of hours spent by a user listening to music and the MUSIC attribute, which described how important music was to that user.



Scatter Plot Between LIST and MUSIC

To handle the missing data, we grouped users based on the 5 values of the MUSIC attribute. For each group, we considered the mean of the LIST value, and generated random values to fill the missing values using a Poisson Distribution, taking care to clip values to 24.
LIST was further grouped and categorized into equal bins of size 4.

- QUESTIONS (ordinal, 0-100):

  The Music habit questions mentioned in the user.csv dataset was marked as Q1 through Q19 in the same order as they appeared. The users had provided ratings in the range of 0-100 which were binned into four equal bin sizes.

Attribute Value	Derived From
Q_POP	Q11, Q12
Q_NEW_MUSIC	Q1, Q2, Q3, Q15, Q17
Q_DANCE	Q7, Q8

Q16 had 6435 missing values, whereas, Q18 and Q19 had 13125. However, all three of them were dropped because they were not so relevant to our objective of rating prediction, and they also did not belong to any of the above groups

c. Merging Users.csv and words.csv to create a meaningful dataset

The Users.csv contained information about the people interviewed, their song preferences, their music habit. On the other hand, the words.csv file contained data about how every user in the users.csv dataset described the different songs they listened to during their interview. It was, thus, of imperative importance to us that we derive some sort of link between these sets of data which would lend some semantic understanding between the users, their music habits and the ratings they provided to artists. We thus created the UserDataProcessed.csv dataset containing all this semantic information. The rationale behind the creation of this new dataset has been described below.

As previously mentioned, every user interviewed (and present in the users.csv dataset) were asked to listen to different song snippets and describe them using the different sets of adjectives provided by the interviewer. We categorized these descriptors into 9 different categories depending upon the sentiment it conveyed (positive, negative, and neutral) and the aspects of the song (lyrics, feel, and personality) it described. Subsequently, we used K-Means clustering

to group all the 50 artists into 5 different clusters/ artist categories based on the average score the artist received corresponding to each word category.

Having categorized all the artists into 5 different categories, we scanned through every user interviewed in the users.csv dataset and generated their response (sentiment) towards different artist clusters. For every user, we observed their response (positive = 1, negative = -1, neutral = 0) to every artist in each of the five categories and generated the mean response to each category. If the response was greater than 0.33 we recorded it as a positive response. Similarly, any response less than -0.67 was recorded as a negative response. All other responses were recorded as neutral.

(Note: response values would always lie between -1 and 1)

The **UserDataProcessed.csv** thus had the following attributes:

- User: anonymized user id
- GENDER (nominal): binary values for Male/Female.
- AGE (ordinal): 0 - 9 bins
- REGION (nominal): regions in UK
- MUSIC (nominal): importance of Music in respondent's life
- A0, A1, A2, A3, A4 (nominal): 5 artist categories. Values are +1 (positive sentiment), 0 (neutral sentiment), -1 (negative sentiment)
- Q_POP (ordinal): feeling towards pop music (0 - 4)
- Q_NEW (ordinal): feeling/habit towards discovering new song (0-4)
- Q_DANCE (ordinal): feeling/ habit towards music essentially for dance, (0 -4)
- LIST (ordinal): 6 bins, hours spent every day to listening to own music and music played in the background.

5. CLASSIFICATION (RATING PREDICTION)

a. Custom-Built Collaborative Filtering Technique:

i. *Creating the proximity matrix:*

The UserDataPreprocessed.csv file is a set of user vectors which has nominal, binary and ordinal attributes. We create a distance matrix for the users based on Simple Matching algorithm between users' vector attributes.

We calculate the distance as follows:

distance = number of non-matches/ number of attributes
= (M01 + M10) / (M00 + M01 + M10 + M11)

We use this distance metric to create a proximity matrix between all the respondents to the interview. We keep a record of the 20 nearest neighbors (lowest 20 scores) for each user in the proximity matrix.

Thus, we now have been able to cluster respondents into groups of 20 based upon their similarity to each other. We use this proximity matrix to predict the rating of an artist's track.

ii. *Classification/ Rating Prediction*

The training data set had 188690 instances of record, and among them we had 13911 instances with no user records in the users.csv dataset. Since this number was only a small fraction of the entire dataset, we decided to drop these records from the training data set.

The rating on the test data set was predicted as follows:

For every record/track in the test dataset, we assign it the average of the rating, derived from the training data set, as provided by the 20 nearest neighbors (obtained from the proximity matrix) of the corresponding respondent in the record.

If the track was not rated by any of the 20 neighbors of the corresponding user in the training data, we assigned the track the average rating provided to the artist of that track by the same 20 users in the training data.

However, if the artist was not rated by any of the same 20 users in the training data, we assigned the track a rating by averaging over all ratings given to all the tracks belonging to all the artists of that same category by those 20 users.

RESULTS

a) Training and testing on entire training dataset:

- Accuracy: 0.731472316468
- Confusion Matrix:

	Predicted: 0	Predicted: 1	Predicted: 2	Predicted: 3
Actual: 0	35607	16229	1101	0
Actual: 1	1410	60400	11847	19
Actual: 2	12	9602	27773	1151
Actual: 3	0	609	4953	4066

b) 70%-30% split (Train - Test):

- Accuracy: 0.467292214975
- Confusion Matrix:

	Predicted: 0	Predicted: 1	Predicted: 2	Predicted: 3
Actual: 0	6235	7713	1613	230
Actual: 1	3031	13293	5227	560
Actual: 2	838	5589	4526	681
Actual: 3	150	939	1361	448

b. Off-the Shelf Classifier: Random Forest Classifier (Python - Sklearn)

We built a Random Forest Classifier using sklearn's ensemble module using a modified training data with the following attributes:

track	sentiment	MUSIC	LIST
HEARD_OF	Label (artist cluster)	Q_POP	REGION
OWN_ARTIST_MUSIC	GENDER	Q_NEW_MUSIC	rating (Predicted)
LIKE_ARTIST	AGE	Q_DANCE	

RESULTS

a) Training and testing on entire training dataset:

- Accuracy: 0.969206826907
- Confusion Matrix:

	Predicted: 0	Predicted: 1	Predicted: 2	Predicted: 3
Actual: 0	52156	666	104	11
Actual: 1	1495	71590	541	50
Actual: 2	212	1696	36542	88
Actual: 3	20	151	348	9109

b) 70%-30% split (Train - Test)

- Accuracy: 0.956688408285
- Confusion Matrix:

	Predicted: 0	Predicted: 1	Predicted: 2	Predicted: 3
Actual: 0	15469	343	54	10
Actual: 1	551	21138	364	25
Actual: 2	74	577	10894	75
Actual: 3	6	47	145	2662

6. CONCLUSION

The EMI music dataset gathered from Kaggle was thoroughly analyzed and preprocessed for our classification task. We predict the rating on a scale of 0-3 (in the Original data the rating is on a scale of 0-100 which has been binned for this project task). We attempted to develop our custom collaborative filtering method which predicts ratings based on the ratings given by other similar users. This method uses the user details to generate a distance matrix and then predicts values based on 'closest' users' ratings. For another approach, we analyzed and selected useful features and fed our modified dataset to an off the shelf classifier (random forest classifier available in sklearn python module). This classifier performed far better than our custom collaborative filter giving an accuracy of 95.7% on a 70-30 data split compared to accuracy of 46.7% of the latter.

7. CONTRIBUTIONS:

The project was done by both of us together with equal contributions. Everyone contributed ideas to develop the feature vector set and customizing the collaborative filtering method for prediction task. The scripts were written together but alternatingly, with one person picking up the pace when the other took a break.