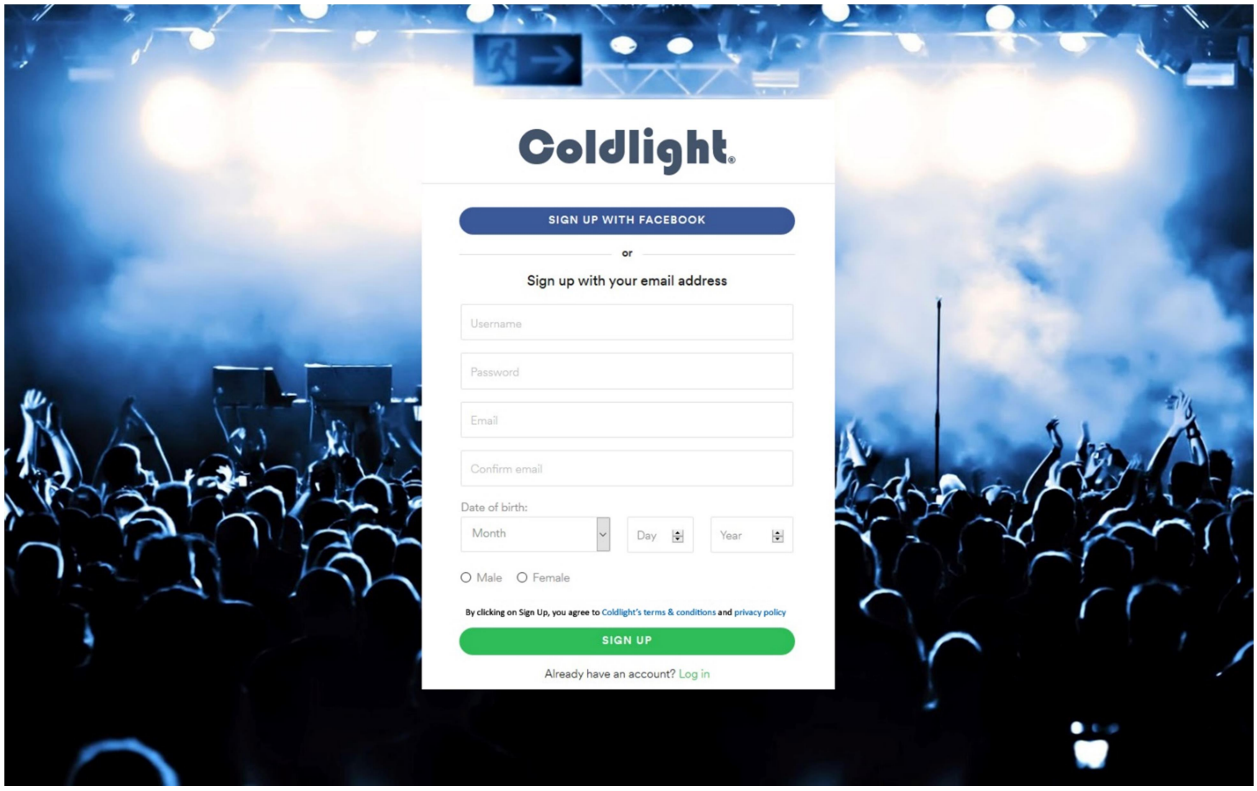


"Music is a cross-cultural universal, a ubiquitous activity found in every known human culture. Individuals demonstrate manifestly different preferences in music, and yet relatively little is known about the underlying structure of those preferences." (Rentfrow et al. 2012).

A group effort by

John Benischeck
Shachi Parikh
Tommy Baw
Xiaomeng Chen
Jordan McIntyre
Chang Wang

Table of Contents

Foreword

The name Coldlight is a fictitious music streaming company we came up with for the purposes of this project. It is not a registered or trademarked name, and any likeliness to a real company is purely coincidental. This is simply meant as a vehicle for discussion and analysis of the EMI music data found on the website Kaggle (full references and credit to the data are given in the bibliography). To that end, any references to partnerships with EMI Records are purely fictitious.

Case Study

Since the introduction of the infamous Napster in 1999, the Internet has become the premier place for music listening. Though Napster was forced to close its doors on its file-sharing business, the precedence for free music was set, with an altered business model. In return for a listener's willingness to hear a few advertisements, streaming companies let users play songs free of charge. This strategy has proved profitable, as music streaming companies such as Songza and Tidal have been valued and acquired for millions of dollars. The key to this model is to grow the base of listeners. [2] [3]

We, the employees of the music streaming service Coldlight, have been tasked with creating a new radio station for first time users of Coldlight that will immediately attract them to use our services. Our company is having significant internet traffic on the landing page, but time spent on our site is short. Users typically leave before trying a demo of our services. We believe that the site, though having much to offer, is being written off by the user. However, for users that spent extended time on our site were significantly likely to return to listen to music at another date. In other words, when given a shot, users seem happy with their experience; it is the first impression that is causing issues.

We want to minimize the probability that a user will be lost by an initial impression, also known as the bounce rate. Working under that principle, it is easier to avoid saying the wrong thing than it is to say the perfect thing. Therefore, we are attempting to minimize playing the wrong type of music to different demographics. To help solve this problem, we will be partnering with EMI Records, who have offered to provide us with some of their user and artist data. A playlist with only approved artists from a selection of those signed to the EMI label (sponsor of Coldlight) will be initially offered to each distinct group of listeners.

EMI did not provide artist names in their datasets. The artist information is de-identified where artists' are labeled 00 to 49. This data also comes with what listeners thought of those artists based on 82 different words. We will approach the project based on the work of Rentfrow et al. by performing a cluster analysis on the descriptive words to determine which words commonly appear with one another via sentiment analysis.[5] Rentfrow et al. publication found underlying factors in music preferences independent of music genre, focusing on emotional and effective reposes instead.[5] Though the artist genres were not included in the data, we will not completely abandon the idea of genre when clustering the words if the sentiment of the words were similar.

Once the clusters are obtained, the next step will be to connect these clusters with different age groups and genders, to determine what they thought of each artist. Although more information than age and gender are provided in the data, we will exclude it. While other streaming services such as Pandora ask listeners for songs or artists they like as a starting point, we want to make the experience of arriving on our site and listening to music simpler and quicker. This practice is common among other streaming sites, such as Spotify.

After we perform a cluster analysis of the words, they will be broken into groups. The counts of the words used to describe each of the artists will allow for the creation of artist profiles. Overall trends in the artist profiles will be looked at by age group and gender. We expect to see a differences between the overall averages and averages of each group's most liked artists. Even if artists do not differ greatly among groups, despite research indicating strong differences by gender and age, they will be kept to allow for the most robust model possible.

To pair artists and users, we will examine the profiles of the top five most liked artist of each age group for both male and female listeners. Based on what the most liked and least like averages for the breakdown of the artist profiles when compared to the overall averages by gender constraints will be made accordingly.

Managerial Report

As you know, Coldlight faces the challenge of acquiring new users, despite our high retention rate of existing users. The main problem we identified is that our sign-up and initialization procedures take too long for new users. They leave our website either during the sign-up phase or during the period of searching for artists to customize their playlists.

To counteract this, we have designed a two-step solution. First, we have streamlined the sign-up process. While we ask for the standard username, password, and email address, we only ask two additional questions: the user's gender, and their birthday. With these two pieces of information, we can build each new listener a custom playlist of music which they are predicted to like. This playlist is available to the user as soon as they log in for the first time, offering gratification much sooner. Thus, the user will increase their time on our site as they become better acquainted with our service.

To build the custom playlists, we first reviewed research regarding the aspects of individual music tastes and preferences. We found that music tastes can be generally broken down by age, as well as gender.[14] For instance, Paul Lamere, a data analyst, looked at 200,000 Spotify users and found more than a 30% difference among males and females in most-liked artists and their genres.[4] Another study found robust age trends in musical taste, where people were broken down into seven age groups.[1] Based on these reports, we chose these two points to be the only questions for our sign-up, as well as our decision model.

These assumptions for our model made our partnership with EMI Records an excellent choice. The data they provided allowed us to view over 32,000 users and their sentiment towards 50 artists. While the data contained many additional users, we chose to focus on the groups who were familiar with the artists they reviewed. Therefore, the dataset we analyzed contained the individual user, their age, gender, a Likert score of the artist, and the words they used to describe that artist.

From this dataset, we clustered similar words into 10 sentiments for each artist broken down by males and females. We further broke down the data into the average sentiment weight for each age group by sex. This was done for the top 5 most liked artist for each age group and sex (reference Figure 3.2 in the Appendix). When compared to the overall sentiment average by gender, we can pinpoint which sentiments matter to each age group. With this, we have built support for our model.

The model, then, is simply to minimize the sum score of sentiment categories least liked for each age group and gender. By minimizing traits least desired by each group, we can populate a custom playlist with artists they will enjoy. To further ensure that the new user likes the playlist, we included stipulations that sentiments important to each age and gender group are emphasized, along with requirements for diversity among artists.

The final result is a set of 14 playlists, broken down by male and female across 7 age groups, as shown below. They are made to consider how each age group and gender feel about music, based on the information they provide us while signing up. Through an easier sign-up, and immediate sampling of our service, we anticipate a higher percentage of site visitor conversions to users. Please refer to the deck in the appendix for updates to the UI.

Technical Report

Our decision support system focuses on how we can minimize the likelihood of putting a disliked artist on a new user playlist, based on that user's age and gender. Specifically, our goal is to select corresponding artists to minimize the total comparative preference rates for the least-liked sentiment of the particular age and gender group. The optimal playlist will be diversified to include artists in different sentiment categories, with a focus on those with higher, desirable sentiments.

Decision Variables:

```
var X{AGE, GENDER, ARTIST} binary;

/* X[i,j,k]=1 indicates that users in the age group i and gender group j
should choose artists k into their playlist.

X[i,j,k]=0 indicates not choose.*/
```

Objective Function:

```
minimizeleast_like=sum {i in ARTIST} preferrate[i,gg,word_p[aa,gg,10]] *
X[aa,gg,i];
* minimize the total comparative preference rate for the least-like category
of the particular age and gender group;
```

In general, we included the following constraints in our model:

- Total artists in the music playlist is 8.

  ```
  con c1:sum{i in ARTIST} X[aa,gg,i]=8;
  ```

- For the three most preferred music sentiments based on the age group and gender, the average comparative preference rate of each sentiment in our playlist should be at least the average level of the preference rate of top 5 preferred artists.

  ```
  conthreelike {r in 1..3}: sum{i in ARTIST}
  X[aa,gg,i]*preferrate[i,gg,word_p[aa,gg,r]]>=average[word_p[aa,gg,r]]*8
  ;
  ```

- For the second least preferred sentiment based on the age group and gender, the average comparative preference rate in this sentiment in our playlist should not exceed the average level of the preference rate of top 5 preferred artists.

  ```
  consecond_hate : sum{i in ARTIST}
  X[aa,gg,i]*preferrate[i,gg,word_p[aa,gg,9]]<=
  average[word_p[aa,gg,9]]*8;
  ```

- For the moderately preferred sentiments (4th to 8th) based on the age group and gender, we should each at least include one artist whose preferred weight in the category is higher than the average level of the category.

```
con middle {r in 4..8}: sum{i in ARTIST}
X[aa,gg,i]*above[i,word_p[aa,gg,r]]>=1;
```

- To ensure diversity, for the ten categories, the total number of artists with high preferred rate based on the age and gender should not exceed 5.

- ```
  conhigh_score {r in 1..10}: sum{i in ARTIST}
  X[aa,gg,i]*above_20[i,word_p[aa,gg,r]]<=5;
  ```

- The total scores based on the age group and gender should be at least 1.2 times the average scores based on the age and gender.

```
conscore_art: sum{i in
ARTIST}X[aa,gg,i]*score_s[aa,gg,i]>=avg_score[aa,gg]*8;
```

A solution to one of the 14 possible outcomes is shown below. For males (column 1) between the age of 13 and 17 (column 2), we include artists 16, 18, 19, 20, 34, 35, 37, 38 (columns 3 and 4) in our playlist. The full solution and support system with UI can be found in the appendix.

[1]	[2]	[3]	X.SOL
1	1	1	0
1	1	2	0
1	1	3	0
1	1	4	0
1	1	5	0
1	1	6	0
1	1	7	0
1	1	8	0
1	1	9	0
1	1	10	0

1	1	11	0
1	1	12	0
1	1	13	0
1	1	14	0
1	1	15	0
1	1	16	1
1	1	17	0
1	1	18	1
1	1	19	1
1	1	20	1

1	1	21	0
1	1	22	0
1	1	23	0
1	1	24	0
1	1	25	0
1	1	26	0
1	1	27	0
1	1	28	0
1	1	29	0
1	1	30	0

1	1	31	0
1	1	32	0
1	1	33	0
1	1	34	1
1	1	35	1
1	1	36	0
1	1	37	1
1	1	38	1
1	1	39	0
1	1	40	0

1	1	41	0
1	1	42	0
1	1	43	0
1	1	44	0
1	1	45	0
1	1	46	0
1	1	47	0
1	1	48	0
1	1	49	0
1	1	50	0

Figure 1.1 – a sample output of the decision support machine

To reach this model, we first had to analyze the datasets given to us by EMI, in accordance with our research findings on how age and gender cause differences in musical taste and preference. We eliminated any values of missing gender, to create the binary decision of male or female. Likewise, we removed any missing ages. Based upon a study carried out by Arielle Bonneville-Roussy from Cambridge's Department of Psychology, we divided age into 7 groups:

$$\begin{matrix} Male \\ Female \end{matrix} \begin{cases} 13-17\ Years \\ 18-24\ Years \\ 25-34\ Years \\ 35-44\ Years \\ 45-54\ Years \\ 55-64\ Years \\ 65+Ye \end{cases}$$

Figure 1.2 – The 14 different input categories
```

Research shows that musical tastes shift as we age are in line with key "life challenges." Teenage years were defined by "intense" music, then early adulthood by "contemporary" and "mellow" as the search for close relationships increases, with "sophisticated" and "unpretentious" allowing us to project status and family values later in life. This study used data from more than a quarter of a million people over a 10 year period.

While reviewing the EMI data, it became apparent that many words were synonyms and that a cluster analysis could be performed to find naturally occurring patterns within the words. A spreadsheet was created giving a total count of words used to describe each artist. From there, a cluster analysis was performed to reduce the amount of similar words. Several clustering methods were performed to confirm we not only had the best fitting dendogram, but also the words in each cluster were similar in nature. Ultimately the Ward Method was used to produce the clusters below (dendogram in appendix…)

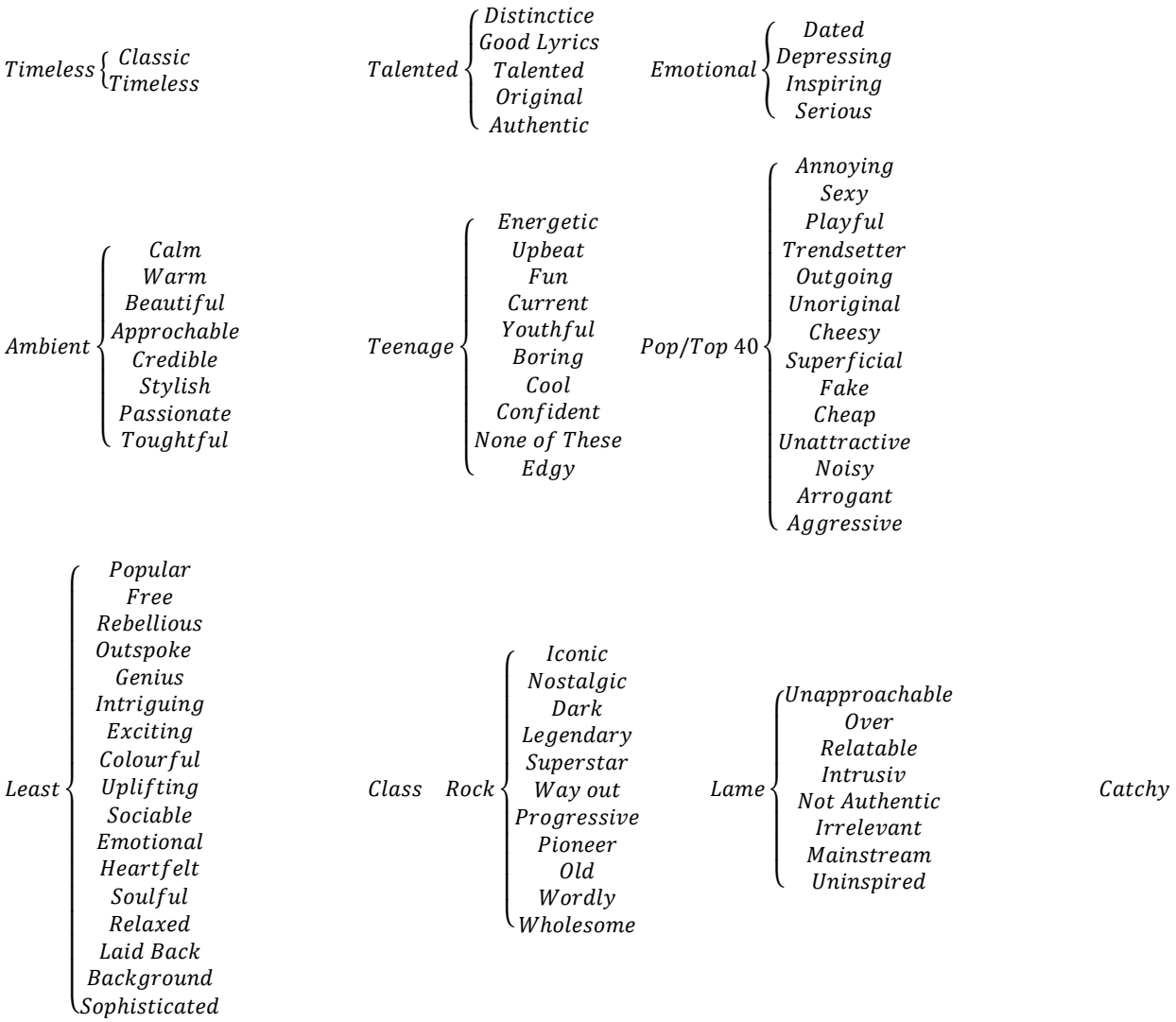


Figure 1.3 – The 10 clusters of words chosen via Ward's method.

Using this sentiment analysis, we were able to name each of the 10 clusters produced from the original 82 words, as seen above. The decision was made to leave catchy as its own cluster as it was the most used word across all artists. For the rest, names were given based on the emotion or music-style that the words described. For example, music that was calm, warm, beautiful, approachable, credible, stylish, passionate, and thoughtful pertained in our opinion to ambient music. The category Least is our largest cluster, and we gave it this name as many of these words were sparingly used in the survey and thus the words in this cluster might not appear to belong together at first glance.

With 10 meaningful clusters, we found the percentage of each cluster used to describe the artists. From this information we can now work towards our preference algorithm. Since the algorithm considers age and gender, we split the data of cluster breakdown per artist to both male and female users.

	A	B	C	D	E	F	G	H	I	J	K
1	art	Timeless	Talented	Emotional	Ambient	Catchy	Teenage	Pop_Top_40	Least	Classic_Rock	Lame
2	0	0.016690341	0.138210227	0.052130682	0.187713068	0.048366477	0.319176136	0.05234375	0.181889205	0.001491477	0.001988636
3	1	0.137333615	0.133636172	0.1513839	0.246461018	0.037185717	0.112719206	0.171561378	0	0	0.009718994
4	2	0.006527415	0.114316797	0.051131419	0.133507398	0.059225413	0.353089643	0.227328111	0	0.000957354	0.053916449
5	3	0.052343666	0.210125496	0.076906575	0.221602488	0.043762737	0.283063392	0.104472809	0	0	0.007722836
6	4	0.134468349	0.290326575	0.077542176	0.193994738	0.068039003	0.181179384	0.047330135	0	0	0.007119641
7	5	0.016221697	0.17303143	0.128759716	0.206150727	0.037174721	0.293342345	0.133491044	0	0	0.01182832
8	6	0.04813108	0.168356375	0.006758833	0.19109063	0.024475166	0.179621096	0.004198669	0.372862263	0.004505888	0

Figure 1.4 -- A sample from the percentage weight of each cluster for the males. Note that One was also created for Females, as well as combined.

We then identify the top five artists by the Likert score users' filled out to see what word clusters users (male or female) are using to describe their favorite artists.

VIEWTABLE: Work.Score

	age_s	gender_s	artist_s	score_s
1	1	1	1	54.53
2	1	1	2	32.64
3	1	1	3	41.07
4	1	1	4	53.12
5	1	1	5	53.59
6	1	1	6	47.93
7	1	1	7	51.27
8	1	1	8	51.86
9	1	1	9	51.18
10	1	1	10	42.6
11	1	1	11	51.45
12	1	1	12	60.09
13	1	1	13	44.87
14	1	1	14	43.41
15	1	1	15	40.7
16	1	1	16	32.5
17	1	1	17	49.77
18	1	1	18	54.09
19	1	1	19	71.92
20	1	1	20	47.74
21	1	1	21	36.54

Figure 1.5score_s is the average Likert score per artist per gender per age

Take, for example, figure 2.1 in the appendix, which only shows a portion of our data for clusters “Timeless” and “Talented”. We will look at the data highlighted in orange where AG1M represents all male users between the ages of 13 and 17 that gave a likability score for artists and they gave artists 18, 28, 29, 11, and 40 the highest average rating. We calculate 0.0157722 from the average weight that all male users used words in the timeless cluster for those 5 artists. If you divide this by the overall weight that male users put into Timeless (0.0560719), we get the table listed as figure 2.2. For AG1M, they place only 28.13% importance towards artists that can be classified as producing timeless music. On the other hand, they enjoy music listed under the “Teenage” category 146.79% of the time compared to other users. Figure 2.2 of the appendix has been color coordinated for ease of use where green represents the category of music that a certain age group puts an emphasis and red for their least enjoyed type of artist. With this dataset (2.2), we have obtained the inputs for our constraints outlined above. Figure 2.3 below shows the full output of our decision support system: 14 artist-optimized playlists, with 8 artists each.

Bibliography

1. Bonneville-Roussy, Arielle, Peter J. Rentfrow, Man K. Xu, Jeff Potter. Music through the ages: Trends in musical engagement and preferences from adolescence through middle adulthood. Journal of Personality and Social Psychology, 2013; 105 (4): 703 DOI: 10.1037/a0033770

2. Crook, Jordan. "Google Buys Songza." *TechCrunch*. AOL Inc, 1 July 2014. Web. 16 Mar. 2016.

3. Etherington, Darrell. "Apple Buys Beats Electronics For $3B." *TechCrunch*. AOL Inc, 28 Mar. 2014. Web. 16 Mar. 2016.

4. Lamere, Paul. "Gender Specific Listening." Music Machinery. N.p., 10 Feb. 2014. Web. 16 Mar. 2016.

5. Rentfrow, Peter J., Lewis R. Goldberg, and Daniel J. Levitin. "The Structure of Musical Preferences: AFive-Factor Model." Journal of Personality and Social Psychology. U.S. National Library of Medicine, June 2011. Web. 16 Mar. 2016.

Special reference and citation to Kaggle.com and EMI music for the artist and user datasets which we used to complete this project. Per the Kaggle.com competition rules, all data will be destroyed after final analysis is submitted. To replicate this experiment, you must register with Kaggle and agree to data terms and conditions to download and use the data files utilized.
(https://www.kaggle.com/c/MusicHackathon)

APPENDIX

Figure 2.1

		Artist1	Artist2	Artist3	Artist4	Artist5	Timeless	Talented
1	AG1M	18	28	29	11	40	0.0157722	0.1888658
2	AG1F	11	26	17	35	40	0.0219846	0.1706131
3	AG2M	49	40	17	11	9	0.0247139	0.1536077
4	AG2F	17	40	19	11	6	0.0175559	0.1685359
5	AG3M	19	4	23	35	22	0.060654	0.2406601
6	AG3F	5	43	44	34	4	0.0589015	0.2468483
7	AG4M	29	7	4	22	34	0.0722811	0.242749
8	AG4F	17	43	8	31	44	0.0261049	0.1807106
9	AG5M	7	22	4	21	6	0.0838473	0.2472319
10	AG5F	15	43	26	29	7	0.0504046	0.215482
11	AG6M	4	22	7	35	49	0.0795477	0.2451082
12	AG6F	29	4	9	36	44	0.0729278	0.206394
13	AG7M	4	43	21	14	23	0.0610645	0.246383
14	AG7F	15	27	30	21	4	0.0498588	0.1929626
15	AGNAM	1	30	26	29	2	0.0506945	0.1977143
16	AGNAF	0	30	28	27	29	0.0292231	0.191369
17	AGNANA	38	23	40	6	48	0.0255178	0.1449652
Average Males							0.0560719	0.22029
Average Females							0.0408702	0.1966144
Total Average							0.0471208	0.2047177
Average over all artists							0.0353614	0.1602547

** Note that this sheet does not contain all 10 clusters and their associated values. This is just a sample dataset.

Figure 2.2

	Timeless	Talented	Emotional	Ambient	Catchy	Teenage	Pop / Top 40	Chang	Classic Rock	Lame
Males 13-17	0.2813	0.8574	0.9751	0.9285	1.1756	1.4679	1.4185	0.3865	0.2888	1.1346
Females 13-17	0.5379	0.8678	0.6458	0.7530	1.2500	1.2857	1.0967	1.2803	1.2874	0.8654
Males 18-24	0.4408	0.6973	0.4542	0.7729	1.5013	1.4643	1.1008	2.3243	0.1349	0.2570
Females 18-24	0.4296	0.8572	0.5929	0.8279	1.1983	1.2187	0.6966	2.4222	0.3276	0.1165
Males 25-34	1.0817	1.0925	1.2098	1.1689	0.8111	0.7878	1.0963	0.3783	1.6365	1.0497
Females 25-34	1.4412	1.2555	1.3567	1.2235	1.0307	0.8418	0.8568	0.0000	0.0000	0.5400
Males 35-44	1.2891	1.1020	0.9419	1.0175	0.7312	0.8044	0.7146	1.2908	1.9787	1.1123
Females 35-44	0.6387	0.9191	1.0143	1.0368	1.2606	1.0182	0.6873	1.8251	0.1590	0.2562
Males 45-54	1.4954	1.1223	0.6569	0.9098	0.9018	0.7346	0.4727	2.3295	1.7797	0.5214
Females 45-54	1.2333	1.0960	1.0037	1.0942	0.7750	0.8885	0.8866	0.9100	2.2399	1.2510
Males 55-64	1.4187	1.1127	0.9313	1.0170	1.0012	0.7535	0.7364	1.2908	1.7319	0.7978
Females 55-64	1.7844	1.0497	0.8045	1.0863	0.8831	0.8757	0.7900	1.0061	1.4094	1.3854
Males 65+	1.0890	1.1184	1.3482	1.2099	1.0789	0.9308	0.8399	0.0000	0.1491	1.0662
Females 65+	1.2199	0.9814	1.1665	0.8855	0.9640	0.9938	1.6374	0.0000	0.8801	1.7591

Figure 3.3

Artist	AG1 M	AG1 F	AG2 M	AG2 F	AG3 M	AG3 F	AG4 M	AG4 F	AG5 M	AG5 F	AG6 M	AG6 F	AG7 M	AG7 F
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0	0	0	0	0	0	0	0
3	0	0	1	0	0	1	0	0	0	0	0	0	1	0
4	0	0	1	0	0	1	0	0	0	0	0	0	0	1
5	0	0	0	0	0	1	0	1	1	0	0	0	0	1
6	0	0	0	0	0	1	0	0	0	0	0	0	0	0
7	0	0	1	1	0	0	1	0	1	1	1	1	0	0
8	0	0	0	0	0	0	1	0	1	0	1	0	0	0
9	0	0	0	1	0	0	0	1	0	0	0	0	0	0
10	0	0	0	1	0	1	1	1	0	1	1	1	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	1	0	0	0	0	0	0	1	1	0
16	1	1	0	0	0	0	0	0	0	0	0	0	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	1	1	1	1	0	0	0	0	1	0	0	0	0	0
19	1	0	0	1	0	1	0	0	0	0	0	0	0	0
20	1	0	0	1	0	1	0	1	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	1	0	0	0	0	0
23	0	0	0	0	0	0	1	0	1	1	1	0	0	0
24	0	0	0	0	0	0	1	0	0	0	1	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	1	0
26	0	0	0	0	1	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	1
29	0	0	0	0	0	0	1	0	1	1	1	1	1	0
30	0	1	0	0	0	0	0	1	0	1	0	1	0	0
31	0	0	0	0	1	0	0	0	0	0	0	1	0	0
32	0	1	0	0	0	0	0	0	0	1	0	0	0	0
33	0	0	0	1	0	0	0	0	0	0	0	0	0	0
34	1	1	1	0	0	0	0	0	0	0	0	0	0	0
35	1	0	0	0	0	0	0	1	0	0	0	1	0	0
36	0	0	0	0	1	0	0	0	0	0	0	0	0	0
37	1	0	0	0	1	0	0	0	0	0	0	0	0	0

38	1	1	0	0	0	0	0	0	0	1	0	0	0	1
39	0	0	0	0	0	0	0	0	0	0	0	0	0	1
40	0	0	0	0	0	0	0	0	0	1	0	0	0	0
41	0	0	1	0	0	0	0	1	0	0	0	0	0	0
42	0	0	1	0	1	0	0	0	0	0	0	0	1	0
43	0	0	0	0	0	0	0	0	0	0	0	0	0	1
44	0	0	0	0	1	0	1	1	0	0	1	0	1	0
45	0	0	0	0	0	0	0	0	1	0	0	0	0	1
46	0	0	0	0	1	0	0	0	0	0	0	0	1	0
47	0	0	0	0	0	0	0	0	0	0	0	0	0	0
48	0	1	0	1	0	0	1	0	0	0	1	1	0	0
49	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50	0	0	1	0	0	0	0	0	0	0	0	0	0	1

SQL Code to pull the datasets

** Note that the dataset “Words” and “user” were initially downloaded from Kaggle.com under the EMI contest. We then added Sex and Age to the Words database via Excel vlookup function. This was done to query datasets by gender and by age, which are discussed in detail above.

```
SELECT Male.Artist, Sum(Male.Uninspired) AS SumOfUninspired, Sum(Male.Sophisticated) AS
SumOfSophisticated, Sum(Male.Aggressive) AS SumOfAggressive, Sum(Male.Edgy) AS SumOfEdgy,
Sum(Male.Sociable) AS SumOfSociable, Sum(Male.[Laid back]) AS [SumOfLaid back],
Sum(Male.Wholesome) AS SumOfWholesome, Sum(Male.Uplifting) AS SumOfUplifting,
Sum(Male.Intriguing) AS SumOfIntriguing, Sum(Male.Legendary) AS SumOfLegendary, Sum(Male.Free)
AS SumOfFree, Sum(Male.Thoughtful) AS SumOfThoughtful, Sum(Male.Outspoken) AS
SumOfOutspoken, Sum(Male.Serious) AS SumOfSerious, Sum(Male.[Good lyrics]) AS [SumOfGood lyrics],
Sum(Male.Unattractive) AS SumOfUnattractive, Sum(Male.Confident) AS SumOfConfident,
Sum(Male.Old) AS SumOfOld, Sum(Male.Youthful) AS SumOfYouthful, Sum(Male.Boring) AS
SumOfBoring, Sum(Male.Current) AS SumOfCurrent, Sum(Male.Current) AS SumOfCurrent1,
Sum(Male.Colourful) AS SumOfColourful, Sum(Male.Stylish) AS SumOfStylish, Sum(Male.Cheap) AS
SumOfCheap, Sum(Male.Irrelevant) AS SumOfIrrelevant, Sum(Male.Heartfelt) AS SumOfHeartfelt,
Sum(Male.Calm) AS SumOfCalm, Sum(Male.Pioneer) AS SumOfPioneer, Sum(Male.Outgoing) AS
SumOfOutgoing, Sum(Male.Inspiring) AS SumOfInspiring, Sum(Male.Beautiful) AS SumOfBeautiful,
Sum(Male.Fun) AS SumOfFun, Sum(Male.Authentic) AS SumOfAuthentic, Sum(Male.Credible) AS
SumOfCredible, Sum(Male.[Way out]) AS [SumOfWay out], Sum(Male.Cool) AS SumOfCool,
Sum(Male.Catchy) AS SumOfCatchy, Sum(Male.Sensitive) AS SumOfSensitive, Sum(Male.Mainstream) AS
SumOfMainstream, Sum(Male.Superficial) AS SumOfSuperficial, Sum(Male.Annoying) AS
SumOfAnnoying, Sum(Male.Dark) AS SumOfDark, Sum(Male.Passionate) AS SumOfPassionate,
Sum(Male.[Not authentic]) AS [SumOfNot authentic], Sum(Male.Field50) AS SumOfField50,
Sum(Male.Background) AS SumOfBackground, Sum(Male.Timeless) AS SumOfTimeless,
Sum(Male.Depressing) AS SumOfDepressing, Sum(Male.Original) AS SumOfOriginal, Sum(Male.Talented)
AS SumOfTalented, Sum(Male.Worldly) AS SumOfWorldly, Sum(Male.Distinctive) AS SumOfDistinctive,
Sum(Male.Approachable) AS SumOfApproachable, Sum(Male.Genius) AS SumOfGenius,
Sum(Male.Trendsetter) AS SumOfTrendsetter, Sum(Male.Noisy) AS SumOfNoisy, Sum(Male.Upbeat) AS
SumOfUpbeat, Sum(Male.Relatable) AS SumOfRelatable, Sum(Male.Energetic) AS SumOfEnergetic,
Sum(Male.Exciting) AS SumOfExciting, Sum(Male.Emotional) AS SumOfEmotional, Sum(Male.Nostalgic)
AS SumOfNostalgic, Sum(Male.[None of these]) AS [SumOfNone of these], Sum(Male.Progressive) AS
SumOfProgressive, Sum(Male.Sexy) AS SumOfSexy, Sum(Male.Over) AS SumOfOver,
Sum(Male.Rebellious) AS SumOfRebellious, Sum(Male.Fake) AS SumOfFake, Sum(Male.Cheesy) AS
SumOfCheesy, Sum(Male.Popular) AS SumOfPopular, Sum(Male.Superstar) AS SumOfSuperstar,
Sum(Male.Relaxed) AS SumOfRelaxed, Sum(Male.Intrusive) AS SumOfIntrusive, Sum(Male.Unoriginal) AS
SumOfUnoriginal, Sum(Male.Dated) AS SumOfDated, Sum(Male.Iconic) AS SumOfIconic,
Sum(Male.Unapproachable) AS SumOfUnapproachable, Sum(Male.Classic) AS SumOfClassic,
Sum(Male.Playful) AS SumOfPlayful, Sum(Male.Arrogant) AS SumOfArrogant, Sum(Male.Warm) AS
SumOfWarm, Sum(Male.Soulful) AS SumOfSoulful

FROM Male
```

GROUP BY Male.Artist

ORDER BY Male.Artist;

Words with Ranking by Artist

SELECT Words.Artist, Words.User, Words.LIKE_ARTIST, Words.Uninspired, Words.Sophisticated, Words.Aggressive, Words.Edgy, Words.Sociable, Words.[Laid back], Words.Wholesome, Words.Uplifting, Words.Intriguing, Words.Legendary, Words.Free, Words.Thoughtful, Words.Outspoken, Words.Serious, Words.[Good lyrics], Words.Unattractive, Words.Confident, Words.Old, Words.Youthful, Words.Boring, Words.Current, Words.Colourful, Words.Stylish, Words.Cheap, Words.Irrelevant, Words.Heartfelt, Words.Calm, Words.Pioneer, Words.Outgoing, Words.Inspiring, Words.Beautiful, Words.Fun, Words.Authentic, Words.Credible, Words.[Way out], Words.Cool, Words.Catchy, Words.Sensitive, Words.Mainstream, Words.Superficial, Words.Annoying, Words.Dark, Words.Passionate, Words.[Not authentic], Words.Field50, Words.Background, Words.Timeless, Words.Depressing, Words.Original, Words.Talented, Words.Worldly, Words.Distinctive, Words.Approachable, Words.Genius, Words.Trendsetter, Words.Noisy, Words.Upbeat, Words.Relatable, Words.Energetic, Words.Exciting, Words.Emotional, Words.Nostalgic, Words.[None of these], Words.Progressive, Words.Sexy, Words.Over, Words.Rebellious, Words.Fake, Words.Cheesy, Words.Popular, Words.Superstar, Words.Relaxed, Words.Intrusive, Words.Unoriginal, Words.Dated, Words.Iconic, Words.Unapproachable, Words.Classic, Words.Playful, Words.Arrogant, Words.Warm, Words.Soulful

FROM Words

WHERE (((Words.LIKE_ARTIST)>="0"));

SAS Code

Initial analysis of data and clustering

```
/*****************************/
/* PART 1 - SIMPLE ANALYSIS  */
/*****************************/


        /*******************************************/
        /* Import the raw datasets, which includes   */
        /* the sumations of words used to describe   */
        /* on a per artist basis.                    */
        /*******************************************/

proc import datafile = "E:\Custom Queries\Excel For SAS\Male Sum of Words by Artist.xlsx"
        DBMS = xlsx
        Out = Male;
run;

proc import datafile = "E:\Custom Queries\Excel For SAS\Female Sum of Words by Artist.xlsx"
        DBMS = xlsx
        Out = Female;
run;

proc import datafile = "E:\Custom Queries\Excel For SAS\Sum Of Words Per Artist Raw.xlsx"
        DBMS = xlsx
        Out = Combined;
run;

proc import datafile = "E:\Custom Queries\Excel For SAS\Sum Of Words Per Artist Raw.xlsx"
        DBMS = xlsx
        Out = transposed_words_raw;
        sheet = 'Words_Transposed';
        run;

/* weights */

proc import datafile = "E:\Output\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = Averages;
        sheet = 'Averages';
        run;
```

```
proc import datafile = "E:\Output\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = Avg_Male;
        sheet = 'Avg_Male';
        run;

proc import datafile = "E:\Output\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = Avg_Female;
        sheet = 'Avg_Female';
        run;

/* artists */

proc import datafile = "E:\Output\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = Combined_Weight;
        sheet = 'Combined';
        run;

proc import datafile = "E:\Output\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = Artist_Male;
        sheet = 'Male';
        run;

proc import datafile = "E:\Output\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = Artist_Female;
        sheet = 'Female';
        run;

        /*****************************************/
        /* Perform a cluster analysis on how close  */
        /* each word is to another based on the     */
        /* artists.                                 */
        /*****************************************/

proc cluster data = transposed_words_raw method = Ward  outtree = Word_Clusters;
var a00 a01 a02 a03 a04 a05 a06 a07 a08 a09 a10 a11 a12 a13 a14 a15 a16 a17 a18 a19 a20
a21 a22 a23 a24 a25 a26 a27 a28 a29 a30 a31 a32 a33 a34 a35 a36 a37 a38 a39 a40 a41 a42
a43 a44 a45 a46 a47 a48 a49;
id words;
```

```
run;

proc tree data = Word_Clusters;
run;


/*************************************************************************
********/
/* PART 2 - Explore the clusters, and the percentage of words in each cluster to the */
/* overall number of words per each artist, as a whole, and by gender            */
/*************************************************************************
********/


	/*************************************************************************
**/
	/* Create a new dataset which sums the clusters of words for each artist */
	/*************************************************************************
**/

	/* Overall */

data Groups_Overall;
	set Combined;
format Group01 6. Group02 6. Group03 6. Group04 6. Group05 6. Group06 6. Group07 6.
Group08 6. Group09 6. Group10 6.;
	Keep Artist Group01 Group02 Group03 Group04 Group05 Group06 Group07 Group08
Group09 Group10 Sums;

		Group01 = Timeless + Classic;
		Group02 = distinctive + good_lyrics + talented + original + authentic;
		Group03 = dated + depressing + inspiring + sensitive + serious;
		Group04 = calm + warm + beautiful + approachable + credible + stylish +
passionate + thoughtful;
		Group05 = catchy;
		Group06 = energetic + upbeat + fun +current + youthful + boring + cool +
confident + none_of_these + edgy;
		Group07 = Annoying +  Sexy + Playful + Outgoing + Unoriginal +cheesy
+superficial + fake + cheap + unattractive + noisy + arrogant + aggressive;
		Group08 = Popular + free+ rebellious +outspoken +genius + intriguing + exciting
+ colourful + uplifting + sociable + emotional + heartfelt + soulful + relaxed + laid_back +
background + sophisticated;
		Group09 = iconic + nostalgic + dark + legendary + superstar + way_out +
progressive + pioneer + old + worldly + wholesome;
```

```
        Group10 = unapproachable + over + relatable + intrusive + not_authentic +
irrelevant + mainstream + uninspired;
        Sums = (Group01 + Group02 + Group03 + Group04 + Group05 + Group06 +
Group07 + Group08 + Group09 + Group10);


Run;

    /* Males */

data Groups_Male;
    set Male;
format Group01 6. Group02 6. Group03 6. Group04 6. Group05 6. Group06 6. Group07 6.
Group08 6. Group09 6. Group10 6.;
    Keep Artist Group01 Group02 Group03 Group04 Group05 Group06 Group07 Group08
Group09 Group10 Sums;

        Group01 = Timeless + Classic;
        Group02 = distinctive + good_lyrics + talented + original + authentic;
        Group03 = dated + depressing + inspiring + sensitive + serious;
        Group04 = calm + warm + beautiful + approachable + credible + stylish +
passionate + thoughtful;
        Group05 = catchy;
        Group06 = energetic + upbeat + fun +current + youthful + boring + cool +
confident + none_of_these + edgy;
        Group07 = Annoying +  Sexy + Playful + Outgoing + Unoriginal +cheesy
+superficial + fake + cheap + unattractive + noisy + arrogant + aggressive;
        Group08 = Popular + free+ rebellious +outspoken +genius + intriguing + exciting
+ colourful + uplifting + sociable + emotional + heartfelt + soulful + relaxed + laid_back +
background + sophisticated;
        Group09 = iconic + nostalgic + dark + legendary + superstar + way_out +
progressive + pioneer + old + worldly + wholesome;
        Group10 = unapproachable + over + relatable + intrusive + not_authentic +
irrelevant + mainstream + uninspired;
        Sums = (Group01 + Group02 + Group03 + Group04 + Group05 + Group06 +
Group07 + Group08 + Group09 + Group10);

Run;

    /* Females */

data Groups_Female;
    set Female;
```

```
format Group01 6. Group02 6. Group03 6. Group04 6. Group05 6. Group06 6. Group07 6.
Group08 6. Group09 6. Group10 6.;
        Keep Artist Group01 Group02 Group03 Group04 Group05 Group06 Group07 Group08
Group09 Group10 Sums;

                Group01 = Timeless + Classic;
                Group02 = distinctive + good_lyrics + talented + original + authentic;
                Group03 = dated + depressing + inspiring + sensitive + serious;
                Group04 = calm + warm + beautiful + approachable + credible + stylish +
passionate + thoughtful;
                Group05 = catchy;
                Group06 = energetic + upbeat + fun +current + youthful + boring + cool +
confident + none_of_these + edgy;
                Group07 = Annoying +  Sexy + Playful + Outgoing + Unoriginal +cheesy
+superficial + fake + cheap + unattractive + noisy + arrogant + aggressive;
                Group08 = Popular + free+ rebellious +outspoken +genius + intriguing + exciting
+ colourful + uplifting + sociable + emotional + heartfelt + soulful + relaxed + laid_back +
background + sophisticated;
                Group09 = iconic + nostalgic + dark + legendary + superstar + way_out +
progressive + pioneer + old + worldly + wholesome;
                Group10 = unapproachable + over + relatable + intrusive + not_authentic +
irrelevant + mainstream + uninspired;
                Sums = (Group01 + Group02 + Group03 + Group04 + Group05 + Group06 +
Group07 + Group08 + Group09 + Group10);


Run;


        /*************************************************************************
*****/
        /* create another dataset that shows the portion of each cluster per artist */
        /*************************************************************************
*****/

        /* Combined */

data Group_Percent_Combined;
        set Groups_Overall;
        drop Group01 Group02 Group03 Group04 Group05 Group06 Group07 Group08
Group09 Group10 Sums;
                Timeless=Group01/sums;
                Talented=Group02/sums;
                Emotional=Group03/sums;
```

```
                Ambient=Group04/sums;
                Catchy=Group05/sums;
                Teenage=Group06/sums;
                Pop_Top_40=Group07/sums;
                Chang=Group08/sums;
                Classic_Rock=Group09/sums;
                Lame=Group10/sums;

run;

        /* Males */

data group_Percent_Male;
        set Groups_male;
        drop Group01 Group02 Group03 Group04 Group05 Group06 Group07 Group08
Group09 Group10 Sums;
                Timeless=Group01/sums;
                Talented=Group02/sums;
                Emotional=Group03/sums;
                Ambient=Group04/sums;
                Catchy=Group05/sums;
                Teenage=Group06/sums;
                Pop_Top_40=Group07/sums;
                Chang=Group08/sums;
                Classic_Rock=Group09/sums;
                Lame=Group10/sums;


run;

        /* Females */

data group_Percent_Female;
        set groups_Female;
        drop Group01 Group02 Group03 Group04 Group05 Group06 Group07 Group08
Group09 Group10 Sums;
                Timeless=Group01/sums;
                Talented=Group02/sums;
                Emotional=Group03/sums;
                Ambient=Group04/sums;
                Catchy=Group05/sums;
                Teenage=Group06/sums;
                Pop_Top_40=Group07/sums;
                Chang=Group08/sums;
```

```
                Classic_Rock=Group09/sums;
                Lame=Group10/sums;

run;


        /* Export the dataset where the users are paird with their age */

        /* Overall */

proc export data=work.Group_Percent_Combined
        outfile="E:\Custom Queries\SAS Output to Excel\Group_Weights.xlsx"
        dbms=xlsx
        replace;
        sheet = "Combined";
        run;

        /* Males */
proc export data=work.group_Percent_Male
        outfile="E:\Custom Queries\SAS Output to Excel\Group_Weights.xlsx"
        dbms=xlsx
        replace;
        sheet = "Male";
        run;

        /* Females */
proc export data=work.group_Percent_Female
        outfile="E:\Custom Queries\SAS Output to Excel\Group_Weights.xlsx"
        dbms=xlsx
        replace;
        Sheet = "Female";
        run;


/*************************************************************************/
/* Using data maximized for building constraints, build the contstraints  */
/*************************************************************************/

proc import datafile = "C:\Users\jdb438\Desktop\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = cccc;
        sheet = 'Female';
        getnames=yes;
        run;
```

```
proc import datafile = "C:\Users\jdb438\Desktop\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = word;
        sheet = 'sheet1';
        getnames=yes;
        run;

proc import datafile = "C:\Users\jdb438\Desktop\Average Cluster Weight by Age Group.xlsx"
        DBMS = xlsx
        Out = average111;
        sheet = 'sheet2';
        getnames=yes;
        run;

data cccc1;
set cccc;
keep Artist;
run;

Proc optmodel;
set Age;
set Gender;
set ARTIST;
set <string> word;

number Artist_female{artist,word};
number average{word};
*number Artist_male{artist,word};

age ={1..8};
gender = {1..2};
artist={1..50};
word={'Timeless','Talented','Emotional','Ambient','Catchy','Teenage','Pop_Top_40','Chang','Clas
sic_Rock', 'Lame'};

var X{Age, Gender,Artist} binary;

read data work.cccc1 into Artist=[artist];
read data work.cccc into [artist] {w in word}< artist_female[artist,w] = col(w)> ;
read data work.average111 into word=[word] average;
*read data Artist_male into artist_male = [Artist] Timeless Talented Emotional Ambient Catchy
Teenage Pop_Top_40 Chang Classic_Rock Lame;
```

```
minimize Age1_Female = sum {i in artist} artist_female[i,'Timeless'] * X[1,1,i];

con c1:sum{i in artist} X[1,1,i]=8;
con c2:sum{i in artist} X[1,1,i]*artist_female[artist,'Classic Rock'>= average['Classic Rock'];


solve;
print X.sol;
quit;
```

Further SAS Code

Constraints and Optimization Model

```
proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx
     Out = average;
     sheet = 'avg_word';
     getnames=yes;
     run;


proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx
     Out = above_avg;
     sheet = 'above_average';
     getnames=yes;
     run;
proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx replace
     Out = preference;
     sheet = 'preference';
     getnames=yes;
     run;
proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx replace
     Out = Score_by_art;
     sheet = 'sheet2';
     getnames=yes;
     run;

proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx replace
     Out = general_preferrate;
     sheet = 'prefer_rate';
     getnames=yes;
     run;

proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx replace
     Out = score;
     sheet = 'score_1';
     getnames=yes;
     run;

proc import datafile = "C:\Users\xc69\Desktop\model.xlsx"
     DBMS = xlsx replace
```

```
      Out = avg_score;
      sheet = 'avg_score_by_group';
      getnames=yes;
      run;

proc import datafile = "\model.xlsx"
      DBMS = xlsx replace
      Out = above_20;
      sheet = 'above_20';
      getnames=yes;
      run;


/*proc standard data=Score_by_art out=Score_by_art_nm replace
print;
run;

proc export data=score_by_art_nm
   outfile='E:\OPR\project\scorebyart.xlsx'
   dbms=xlsx;
 run;*/

Proc optmodel;
set AGE;     *AGE groups from 1-7;
set GENDER; * Gender groups 1 and 2, 1 means male and 2 means
female;
set ARTIST; *Represents 1-50 artists;
set <string> WORD; * Represents 10 word categories;
AGE={1..7};
GENDER= {1..2};
ARTIST={1..50};
WORD={'Timeless','Talented','Emotional','Ambient','Catchy','Teen
age','Pop_Top_40','Chang','Classic_Rock', 'Lame'};

set <num, num, num> PREFER;
*PREFER[age,gender,rank] to represent the ranked preference for
age/gender;

set <num, num, str> LIKE;
*LIKE[age,gender,word] to represent the preference rate each
age/gender group assigned to word category;

set <num,num,num> SCORE;
*SCORE[age,gender,artist] to represent scores each age/gender
group assigned to each artists;

set <num,num> AGEGENDER;
```

```
*AGEGENDER[age,gender] to represent age and gender group;

number aa=1;
*the user's age information, assigned to different age group
from 1 to 7;

number gg=1;
*the user's genger information, 1 means male, 2 means female;

number average{WORD};
*the general average comparative preference rate for each word
categories;

number above{ARTIST,WORD};
*the artists who have higher comparative preference rate
compared to the average preference rate;

number above_20{ARTIST,WORD};
*above_20[i,S]=1 means the artists i have a more than 1.2 times
the average preference rate for word category S;
*above_20[i,S]=0 means the artists i's S category preference
rate is less than 1.2 times the average preference rate of this
category;

string word_p{PREFER};
*word_p[i,j,r]means the r_th preferred word categories for age
group i and gender group j;

num prefer_rate{LIKE};
*likerate[i,j,w]means the comparative preference rate for the w
category for age i and gender j;

num score_s{SCORE};
*score_s[i,j,k]means the score age i and gender j assigned to k
artist;

num avg_score{AGEGENDER};
*avg_score[i,j] represent the average score that age i and
gender j assigned to 50 artists;

var X{AGE, GENDER, ARTIST} binary;
/* X[i,j,k]=1 indicates that users in the age group i and gender
group j  should choose artists k into their playlist.
   X[i,j,k]=0 indicates not choose.*/

read data work.average into WORD=[words] average;
```

```
read data work.above_avg into ARTIST=[art]{w in WORD}<
above[art,w] = col(w)>;
read data work.above_20 into ARTIST=[art]{w in
WORD}<above_20[art,w]=col(w)>;
read data work.preference into PREFER=[age_p gender_p
rank_p]word_p;
read data work.general_preferrate into LIKE=[art_l gender_l
word_l]prefer_rate;
read data work.score into SCORE=[age_s gender_s
artist_s]score_s;
read data work.avg_score into AGEGENDER=[age_a
gender_a]avg_score;

minimize  least_like=sum {i in ARTIST}
prefer_rate[i,gg,word_p[aa,gg,10]] * X[aa,gg,i];
* minimize the total comparative preference rate for the least-
like category of the particular age and gender group;

con c1:sum{i in ARTIST} X[aa,gg,i]=8;
* the total artists in the playlist is 8;

con threelike {r in 1..3}: sum{i in ARTIST}
X[aa,gg,i]*prefer_rate[i,gg,word_p[aa,gg,r]]>=average[word_p[aa,
gg,r]]*8;
* for three-most like word categories, the average comparative
preference rate should be at least the average top 5 preference
rate;

con second_hate : sum{i in ARTIST}
X[aa,gg,i]*prefer_rate[i,gg,word_p[aa,gg,9]]<=
average[word_p[aa,gg,9]]*8;
* for the second-hate like word categories, the average
comparative preference rate should not excess the average top 5
preference rate;

con middle {r in 4..8}: sum{i in ARTIST}
X[aa,gg,i]*above[i,word_p[aa,gg,r]]>=1;
* for each of the moderate word categories, we should at least
choose one artist whose word category is higher than average
rate;

con high_score {r in 1..10}: sum{i in ARTIST}
X[aa,gg,i]*above_20[i,word_p[aa,gg,r]]<=5;
* for the 1.2 higher than preference rate, we should not include
more than 5 artists to ensure diversity;
```

```
con score_art: sum{i in
ARTIST}X[aa,gg,i]*score_s[aa,gg,i]>=avg_score[aa,gg]*8;
* the average score in our playlist should be at least the
average score of the age/gender group;

solve;
print X.sol;
quit;
```

HTML Code for User Interface

Note that this is a sample of the code to direct to a specific user. The code is repetitive if you want to link each of the 14 types to their respective artist list.

Page 1

```
<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true"
   CodeFile="Default.aspx.cs" Inherits="_Default" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">
</asp:Content>
<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">
   <h2>
      Welcome to ColdLight!
   </h2>
   <h1 style="color: #000000">
    <asp:Label ID="Label1" runat="server"   ForeColor="Black">Gender</asp:Label><br />
      <input type="radio" name="Gender" value="Female" /> Female
      <input type="radio" name="Gender" value="Male" /> Male
   </h1>
   <h1 style="color: #000000">
      <asp:Label ID="Label2" runat="server" Text="Select Your Age" ForeColor="Black"></asp:Label><br
/>
         <input type="radio" name="Age" value="13-17" /> 13-17<br />
         <input type="radio" name="Age" value="18-24" /> 18-24<br />
         <input type="radio" name="Age" value="25-34" /> 25-34<br />
         <input type="radio" name="Age" value="35-44" /> 35-44<br />
         <input type="radio" name="Age" value="45-54" /> 45-54<br />
         <input type="radio" name="Age" value="55-64" /> 55-64<br />
         <input type="radio" name="Age" value="65 and Above" /> 65 and Above<br />
      <asp:Button ID="Button1" runat="server" Text="Welcome!" BackColor="#CCCCCC"
      ForeColor="Black" />
   </h1>
</asp:Content>
```

Page 2

```
<%@ Page Title="Home Page" Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true"
   CodeFile="Default.aspx.cs" Inherits="_Default" %>

<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">
</asp:Content>
<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">
   <h2>
      Welcome to CODELIGHT!
```

```
  </h2>
  <h1>
    <asp:Label ID="Label1" runat="server" Text="Label">Gender : </asp:Label>
    <asp:Label ID="Label2" runat="server" Text="Label">Female</asp:Label><br />

    <asp:Label ID="Label3" runat="server" Text="Label">Age : </asp:Label>
    <asp:Label ID="Label4" runat="server" Text="Label"> 25-34</asp:Label><br /><br />

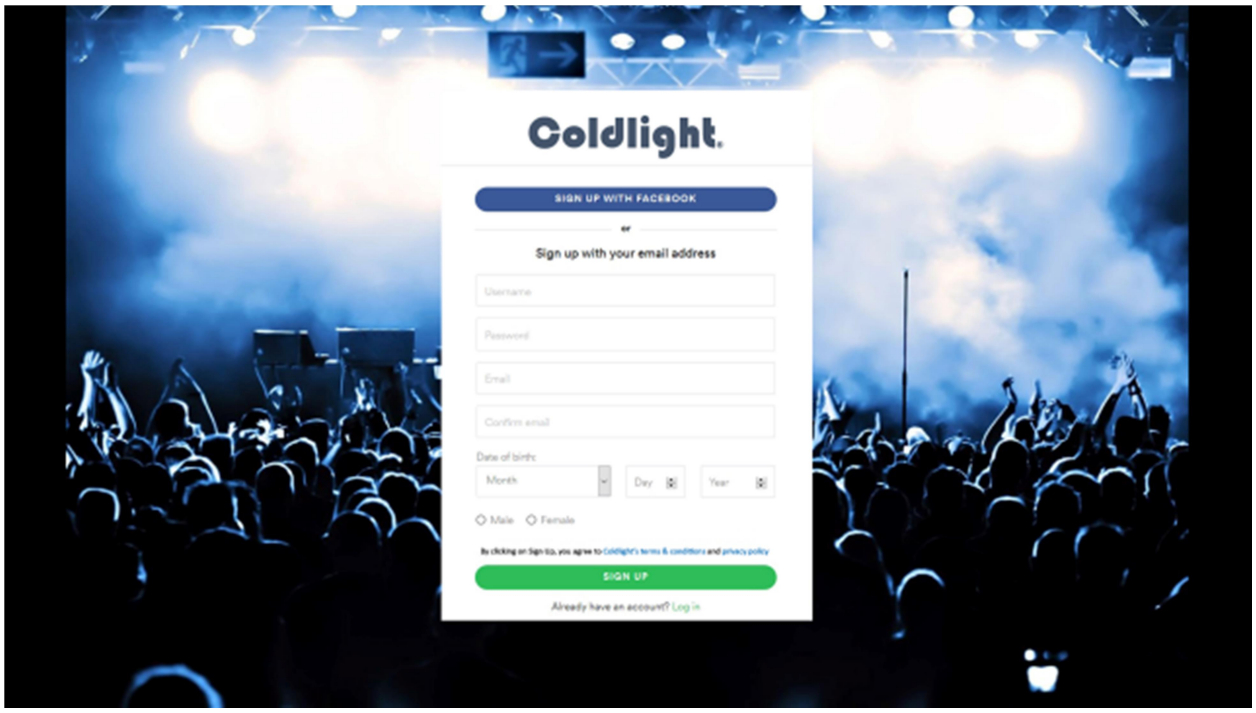
    <asp:Label ID="Label27" runat="server" Text="Label">The Following Artists have been Selected For
You!</asp:Label><br /><br />

    <asp:Label ID="Label5" runat="server" Text="Label">Artist1 </asp:Label>
    <asp:Label ID="Label6" runat="server" Text="Label">Artist2 </asp:Label>
    <asp:Label ID="Label7" runat="server" Text="Label">Artist3 </asp:Label>
    <asp:Label ID="Label8" runat="server" Text="Label">Artist4 </asp:Label>
    <asp:Label ID="Label9" runat="server" Text="Label">Artist5 </asp:Label>
    <asp:Label ID="Label10" runat="server" Text="Label">Artist6 </asp:Label>
    <asp:Label ID="Label11" runat="server" Text="Label">Artist7 </asp:Label>
    <asp:Label ID="Label12" runat="server" Text="Label">Artist8 </asp:Label>
    <asp:Label ID="Label13" runat="server" Text="Label">Artist9 </asp:Label>
    <asp:Label ID="Label14" runat="server" Text="Label">Artist10 </asp:Label>
    <asp:Label ID="Label15" runat="server" Text="Label">Artist11 </asp:Label><br />

    <asp:Label ID="Label16" runat="server" Text="Label">1</asp:Label><span style="padding-
left:52px;"></span>
    <asp:Label ID="Label17" runat="server" Text="Label">0</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label18" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label19" runat="server" Text="Label">0</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label20" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label21" runat="server" Text="Label">0</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label22" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label23" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label24" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label25" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>
    <asp:Label ID="Label26" runat="server" Text="Label">1</asp:Label><span style="padding-
left:50px;"></span>

  </h1>
</asp:Content>
```

UI Mockup



** Note that this is not how the HTML code will look. This is the ideal front-end, user-facing part of the System. The HTML code is a more basic front-end, with output of the optimized playlist.