

BACHELOR THESIS
COMPUTER SCIENCE



RADBOUD UNIVERSITY

Analysing Spotify listening data from Armada Music

Author:

Randy Wanga
s4462939

First supervisor/assessor:

Prof.dr.ir, Arjen P. de Vries
a.devries@ru.nl

Second supervisor:

Dhr, Paul Kooij
paul.kooij@armadamusic.com

Second assessor:

Prof.dr, Martha A. Larson
m.larson@cs.ru.nl

August 21, 2018

Abstract

This Bachelor's thesis uses the Spotify listening data obtained from Armada Music to perform data analysis. We analyse this data with the help of two different APIs. We develop a classification for tracks using k-Means clustering, and use this grouping to perform more analysis on the data. We show that we can gain meaningful knowledge from this data that can be used to optimise track performance.

Contents

1	Introduction	2
2	Preliminaries	4
2.1	Tableau	4
2.2	k-Means clustering	4
2.3	Linear Support Vector Classification	5
2.4	F1-scores	5
3	Research	6
3.1	Data sets	6
3.1.1	Spotify Analytics API	6
3.1.2	Spotify Public API	8
3.2	Investigative Analysis	9
3.2.1	Analytics API	9
3.2.2	Full data set	14
3.3	Classification analysis	15
3.4	Skiprate analysis	19
4	Related Work	21
5	Conclusions	22

Chapter 1

Introduction

Armada Music is a Dutch electronic music record label founded by Armin van Buuren, Maykel Piron and David Lewis in 2003. Since its inception it has grown to become an award-winning label, being crowned Best Global Record Label for five consecutive years. It is currently the largest dance music label in the world, spanning 87 signed artists and 54 active sublabels. Armada Music offers their music on various music streaming platforms, such as Spotify, Deezer and Apple Music. With a catalogue of their scope, they have stored copious amounts of data. This data is stored on a Google Big-Query server; however, up until this research, this data was merely stored and not analysed. We look at this data to learn more about Armada's demographic.

For Armada, gaining insight into what their demographic likes listening to and investigating their listening behaviour can help influence marketing decisions, and this is exactly what we are trying to accomplish with this research. Since no one else has previously analysed this data to extract any meaningful listener data from it, Armada has potentially missed out on a lot of opportunities to advertise tracks, artists and albums at the right times, and subsequently generated less revenue than they could have. Therefore, it is in their best interest to do more with data than just storing it.

Our research has been split up into two parts. Firstly, we explore a case study to investigate how powerful their data can be. To do this, we first explore the data to see what we are working with. Then, we take a sample of this data so we can efficiently work with it. By visualising different attributes from the sample, we can easily discover behavioural patterns. Secondly, we dive deeper into the data to discover information that is not visible at first glance. We use two Spotify APIs to extract different data from Armada's records, and join these two databases. Most notably, we take a look at four machine-learned attributes that define the properties of

a track, and use these to generate a classification based on these attributes. To create classes, we use k-Means clustering, where $k = 4$. We then train an SVM classifier on the data. To validate this classifier, we test it by computing its F1-score. We do this to determine the accuracy of our classifier. With this classifier in hand, employees at Armada can predict the classification of tracks that were not present in the test set and employ this system on a scope beyond this research. We import our newly generated classes back into our database and perform further analysis, looking at countries separately. We then draw conclusions from both parts of the research.

Prior research has looked into the cultural aspects of music preferences and how this differs per region[6]. Notably, the mainstreaminess aspect of music differs per country[5]. Armada Music runs a lot of social media accounts and this includes national social media websites like for example vk.com, which is a Russian social media platform. Armada's posts here are made exclusively in Russian. Gaining insight in tracks that perform well in specific countries, e.g. Russia, can help Armada in tailoring campaigns for tracks to the right audiences. Using our generated classes, we can look at differences in listening behaviour for said classes per country, to aid in this task.

Chapter 2

Preliminaries

2.1 Tableau

Tableau is a big data management, analysis and visualisation tool geared towards business intelligence. It can query various types of databases and generate graphs based on these queries. For this thesis, Tableau is useful since it is capable of easily showing data very clearly. While it was provided to us, we have decided to use this in favour of other methods because of some key features:

- The ability to easily perform join and union operations on datasets.
- A simple interface allows to quickly try out different kinds of visualisations using different dimensions and attributes.
- There are a couple of built-in analysis tools so we did not always have to implement algorithms ourselves.
- Databases can be extended with computations with Tableau.

2.2 k-Means clustering

k-Means is an algorithm used to generate a clustering in a dataset. The algorithm can be described as follows:

1. The algorithm initialises and chooses k random observations as initial centroids of the clusters.
2. The algorithm then switches back and forth between the two following steps:
3. Every entry in the dataset is assigned to a cluster by computing the Euclidean distance of the entry to the centroids and choosing the closest cluster.

4. The centroids are then set to the mean of every cluster.
5. Once entries are no longer assigned to new clusters, the algorithm finishes.

In our thesis, we have used k-Means clustering twice. k-Means was first published by J. MacQueen in "Some Methods for Classification and Analysis of Multivariate Observations" [4].

2.3 Linear Support Vector Classification

In this thesis we use a Support Vector Machine (SVM for short) to train a classifier on our data. It does this by assigning each point to one of two classes. It does this in such a way that there is a clear separating gap between the points in the two classes. In more detail, given a training set $\dots, (x_i, y_i), \dots$, we compute the hyperplane such that it divides the $(x_i, 0)$ from the $(x_i, 1)$ with minimal loss. Of course, since our problem is a multi-class problem, we must reduce the multi-class problem into multiple binary-class problems [2].

2.4 F1-scores

To measure the validity of our clustering and effectiveness of our classifier, we test it by computing the F1-score of the classifier. It makes use of the precision and recall of the classifier. The precision of a classifier is defined as the number of true positives divided by the total number of positives. Recall is defined as the number of true positives divided by the number of true positives and false negatives. As formulas:

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}$$

$$\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$$

Both of these give a number between 0 and 1. The closer the outcome is to 1, the better the classifier performs.

The F1-score is then computed as follows:

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

Just like precision and recall individually, this results in a score between 0 and 1. Similarly, the closer this score is to 1, the better the classifier performs [1].

Chapter 3

Research

Our case study consists of an exploratory data analysis on listening data from online music services, provided to us by Armada Music. The main goal of this research is to extract as much useful knowledge from this data as possible. This knowledge can then be used to start better campaigns that generate more traffic, and therefore more revenue. As such, our main research question is the following one, and it is split into two parts:

”What can we learn from Armada Music’s Spotify data and how can we use this knowledge to optimize track performance?”

To start answering this question, we first need to explore the data to obtain a better understanding of what we are working with.

3.1 Data sets

The Armada Music dataset consists of listening data provided by Spotify. When a user listens to a track on Spotify for more than 30 seconds, that stream is saved in the database. The database consists of CSV and JSON files.

3.1.1 Spotify Analytics API

Initial inspection of the data was done directly on the file system. From these files we can gather a general idea what kind of data is available to us and we can start theorizing various metrics we can pull from this data.

The data is split into three separate categories. Every separate day of streams has files matching these three categories: track data, user data and stream data. A lot of attributes exist that we did not use, so we only list the important attributes in table 3.1:

Table 3.1: Data attributes Spotify Analytics API

Attribute	Part of	Additional comments
Track ID	Track/Stream data	
ISRC	Track data	International standard for record identification.
Track name	Track data	
User ID	User/Stream data	
Country	User data	
Birth year	User data	
Gender	User data	
Timestamp	Stream data	
Length	Stream data	Length of the particular stream.
Source	Stream data	Indicates if the stream came from a user's own playlist, somebody else's playlist, a full album, from the artist page's, using the search module or other source, for instance an embedded player in a browser.
Device Type	Stream data	Desktop, mobile or tablet.
OS	Stream data	Windows, OSX, Linux, Android or iOS.

Because the stream data contains both the User ID and Track ID, we can use these variables to link the stream data to user and track data, respectively. This way, we create one larger dataset with every possible variable.

An interesting observation arises when we compare the CSV and JSON files: they do not contain the same attributes. For instance, the country attribute is missing from the JSON files while it is present in the CSV files. Thankfully, this does not pose a problem, since all the attributes we need are present in the CSV files, so we perform our analysis on those files.

3.1.2 Spotify Public API

Apart from the Analytics API, Spotify also has a Public API with different information of all their tracks. We use two features of this API: "Get a Track" and "Get Audio Features for a Track". Like with the Analytics API, we will only consider relevant attributes:

Table 3.2: Data attributes Spotify Public API

Attribute	Part of	Additional comment
ISRC	Track data	International standard for record identification.
Danceability	Audio Features	Measure for how well one can dance to the track, based on tempo and rhythmic stability.
Energy	Audio Features	Measure for the perceived intensity and energy of the track, based on speed, loudness and noisiness.
Instrumentalness	Audio Features	Measures if a track is instrumental or vocal. An instrumentalness value of above 50% should indicate an instrumental track.
Valence	Audio Features	Measure for the positiveness of a track. A higher valence suggests a bright and cheerful track, whereas a lower valence suggests a gloomy and moody track.

To combine this track data with the track data from the Analytics API, we use the ISRC attribute to join the two sets. Though a track ID attribute exists in the Public API, we have opted not to use it, since the track IDs in the Analytics API appear to be hashed, whereas the track IDs in the Public API are not. We refer to this complete, joined data set as the full data set. We take a sample of this data from February 26th to March 4th of 2018, once again to reduce computation time in Tableau.

3.2 Investigative Analysis

3.2.1 Analytics API

Using Tableau and a sample of the data gathered from the Analytics API, we can create a dataset utilising unions to stitch together data from different days. This sample was acquired by taking 7 days' worth of stream files from the USA, from November 10th to November 17th of 2017. We have opted for this smaller sample because Tableau queries over CSV files are inefficient, most likely due to how long they take to parse in Tableau. This problem is made worse when we join or union more CSV files. A smaller sample circumvents this problem. To connect the stream database with the track and user databases, we join on the `track_id` and `user_id` variables, respectively. This way, we obtain one large file with all of the databases in one large dataset.

Figure 3.1: Analytics API: Histogram of birth years

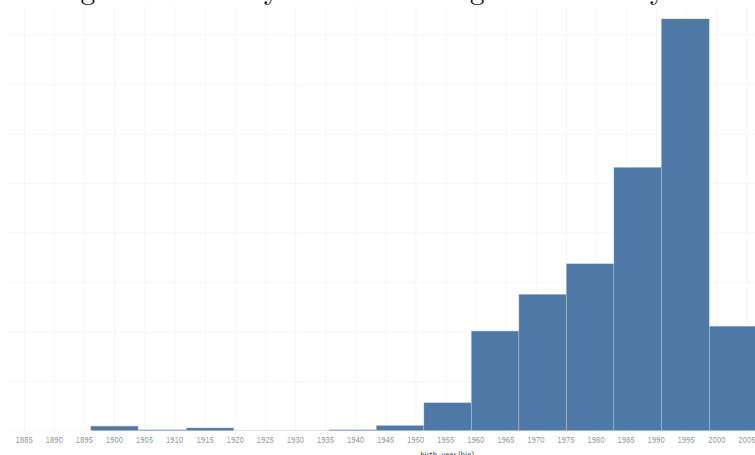
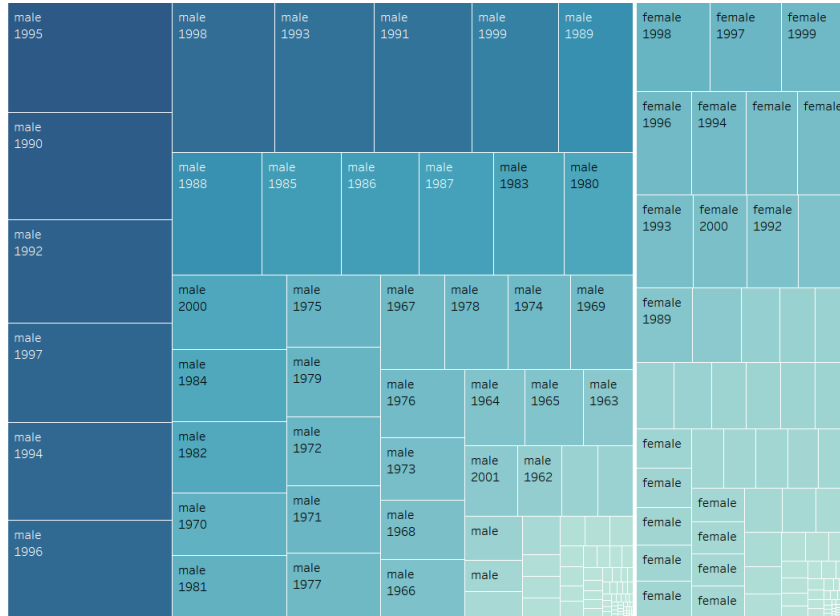
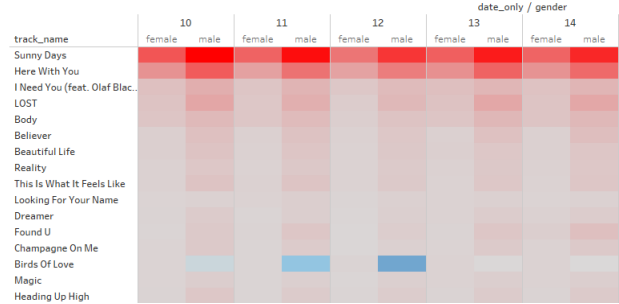


Figure 3.2: Analytics API: Treemap of demographic



A treemap of the US demographic can be seen in Figure 3.2. Treemaps are nested, hierarchical visualisations of data. The data is represented as a tree, and every branch is represented by its own block in the graph. These are then divided into sub-branches based on a chosen hierarchy.[7] In figure 3.2, we have two branches: male and female. The hierarchy is based on the percentage of occurrences of birth years, so the most common birth years will be represented by the largest and darkest coloured tiles in the block. This showcases the most common birth years more clearly, while also separating the data based on gender. From this, we can conclude the male demographic is, on average, slightly older than the female demographic. Additionally, the male demographic is much larger in comparison. In the actual Tableau sheet, this map also shows percentages in every leaf.

Figure 3.3: Analytics API: Gendered Listening Table



The gendered listening table in Figure 3.3 shows the play percentages of a track spread out over the various days in this sample and divides those in male and female plays.¹ The table is colour coded as follows: more red indicates a higher play percentage, and more blue indicates a lower play percentage. A dimmed colour suggests a play percentage that lies somewhere near the mean. Using this knowledge, the highlight table should reveal that while most tracks are predominantly listened to by men, Birds Of Love is either more popular with women or less popular with men. This hints at a difference in listening behaviour between men and women, which we will explore in more detail in section 3.3.

¹These percentages have been censored out, and are visible in the actual Tableau sheet.

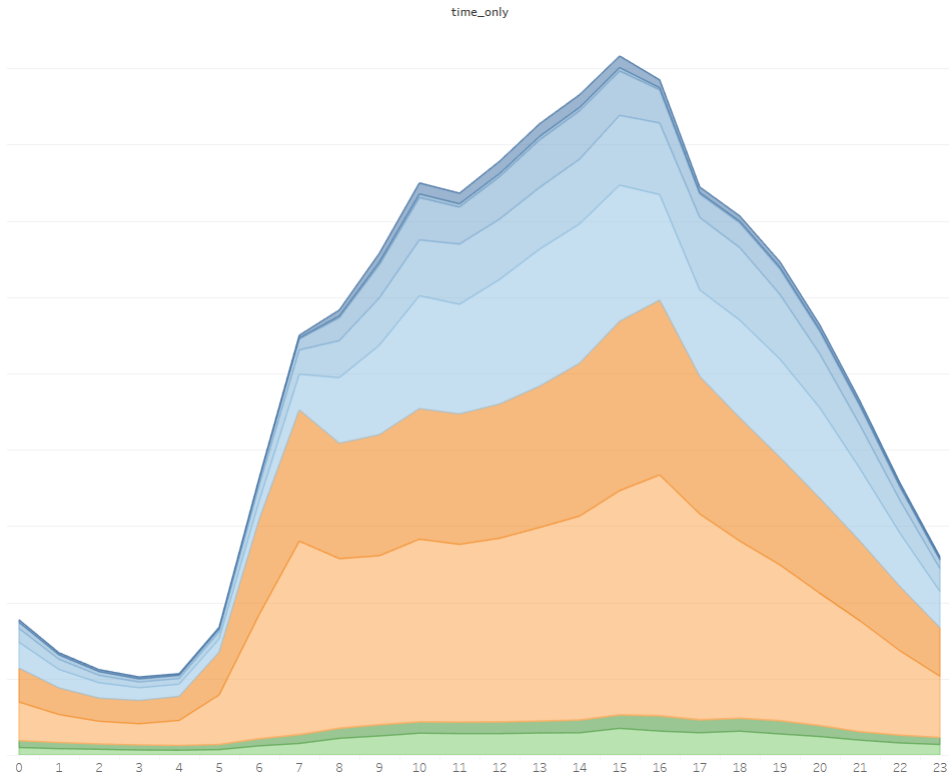


Figure 3.4: Analytics API: Count of device types and operating systems per hour

Figure 3.4 displays the cumulative count of different device types and operating systems per hour. This figure displays the average of the week of this sample data, but each individual day showed a similar graph. Blue represents desktop devices, orange represents mobile devices and green represents tablets. Judging from the graph, 3 pm is the most popular time of day to listen to music, whereas 4 am is the least popular time to do so. Mobile iOS devices (beige) make up the largest part of the population, with Android (darker orange) following in second place. Desktop users are divided over the various operating systems, though Windows (lightest blue) takes the largest share. A notable observation is that the peak hour for mobile devices is an hour later than for desktop devices: 4 pm vs 3 pm. There is also a noticeable peak around 7 am. A possible explanation for these peaks would be that these are common commute hours and people listen to music while travelling to and from work.

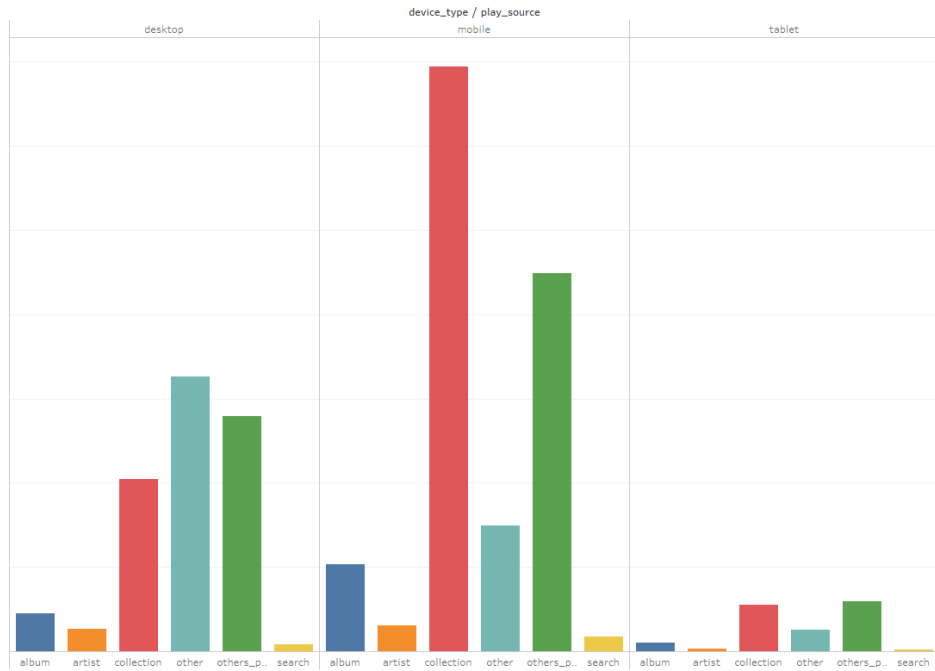


Figure 3.5: Analytics API: Distribution of play sources over devices

In Figure 5, we highlight a notable difference in play sources on different devices. It can be seen that users favour their own collection of tracks when on their mobile devices, but are more likely to listen to other people's playlists on desktop. Since mobile playlist management can be a hassle, people might just favour to listen to their own collection of music when on a mobile device for the sake of convenience. Alternatively, people might be more interested in their own collection of music while commuting versus when sitting in a workplace or at home. Another consideration is the cost of data on mobile. Since users can save their playlists on their phone for offline listening, people might opt to listen to these offline playlists to save data. Desktop users also listen more to other sources, for example embedded tracks. This is most likely because embeds are most common on websites that are typically accessed with a desktop computer.

3.2.2 Full data set

Now we take a look at four of the track attributes: danceability, energy, instrumentalness and valence. Using these four we can create an interesting clustering:

Figure 3.6: Full data set: Clustering of countries based on machine-learned properties

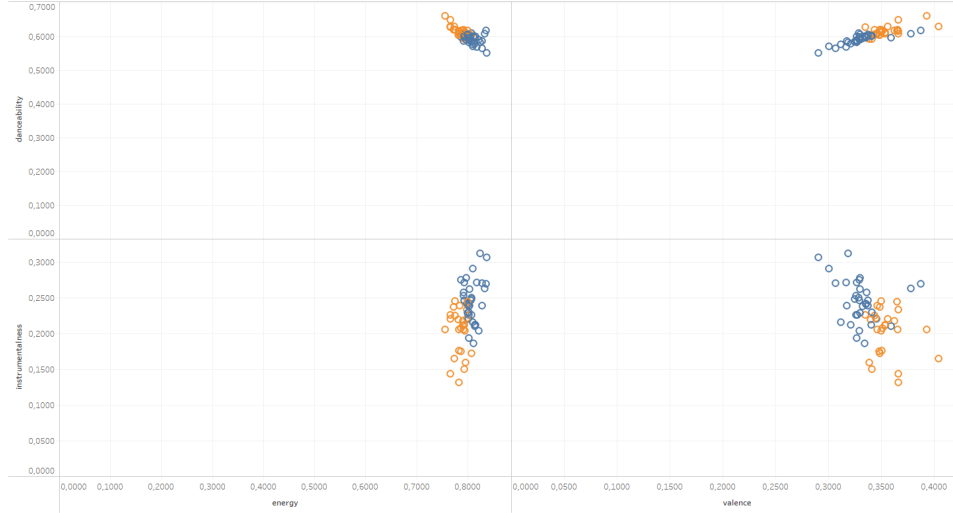


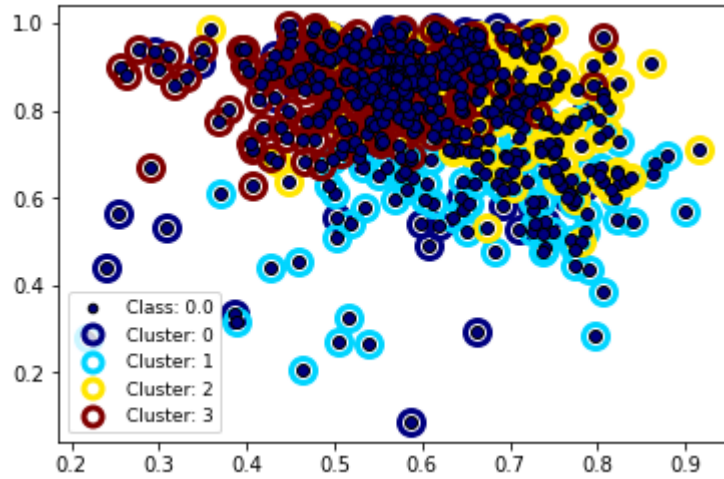
Figure 3.6 shows a scatter plot of the averages of the aforementioned four variables. Each point in the figure represents a country and their average danceability/energy/instrumentalness/valence.² We use k-Means clustering, where $k = 2$, to obtain these clusters. We have chosen 2 clusters here since there are few data points, namely 26, and trying to cluster it into more, smaller clusters would be pointless. Looking at the two clusters we can easily see these variables are in some way connected. Countries in cluster 1 (the blue cluster) listen to tracks with lower danceability and valence but higher instrumentalness and energy than countries in cluster 2. Having found a correlation is a good start, but grouping tracks into two groups would be underfitting the data. If we can create more relevant groups to separate the tracks, we will be able to more accurately adjust marketing strategies.

²Computing the median instead of the average could be a better measure, since it protects against outliers. This was not tested because of time constraints.

3.3 Classification analysis

Using the four attributes of danceability, energy, instrumentality and valence, we can classify the tracks into different groups. However, we lack an actual class attribute, which means we will have to create our own. To do so, we extract the 500 most streamed tracks from the database and perform k-Means clustering on them in Python, based on our four attributes. For clarity, we have chosen $k = 4$, so we will create a multi-class dataset with four classes. Since we are looking at four attributes, choosing a k of 4 seems like a logical decision. We have not tested other values of k due to time constraints.

Figure 3.7: Full data set: Clustering of 500 top tracks based on machine-learned attributes



Every dot in Figure 3.7 represents a track, and its outline represents the cluster to which the track belongs. While the graph is difficult to parse with just the human eye, we can still observe a separation between clusters, despite the noise. However, there exists uncertainty whether or not this clustering is truly valid. Unfortunately, since we used the clustering to generate classes, we cannot compute the cluster validity of our clustering. After all, we have no existing classes to match our clustering against. As a workaround, we train a multi-class one vs rest classifier using Linear Support Vector Classification. Using this classifier allows new tracks that were not present in this test set to easily be classified and added to this database. Aside from this, we can use the classifier to test our clustering, and see if the trained classifier produces largely the same result. This would mean our Support Vector Machine agrees with the clusters, and can correctly distinguish these clusters.³

³We did not further inspect these support vectors.

On this particular instance of k-Means, our classifier gives us the following test scores:

Table 3.3: Accuracy classifier

	Group 0	Group 1	Group 2	Group 3	Average/Total
Precision	0,96	1,00	0,97	0,93	0.96
Recall	0,98	0,80	0,99	0,99	0.96
F1-score	0,97	0,89	0,98	0,96	0.95
Support	90	88	145	177	500

Our classifier reaches an F1-score of 0.95, which implies our classifier can mostly correctly assign the tracks into the right groups, according to the clustering. For the purpose of this research, this makes our cluster suitable for analysis, since we want to take a look at differences in listening behaviour based on attribute separation. A notable observation: Groups 2 and 3 have a higher support than groups 0 and 1, which means they contain relatively more tracks.

Now that we have classified every track, we can import these classifications back into Tableau. Here we can examine the values for every group to see what differentiates them from each other:

Table 3.4: Class properties

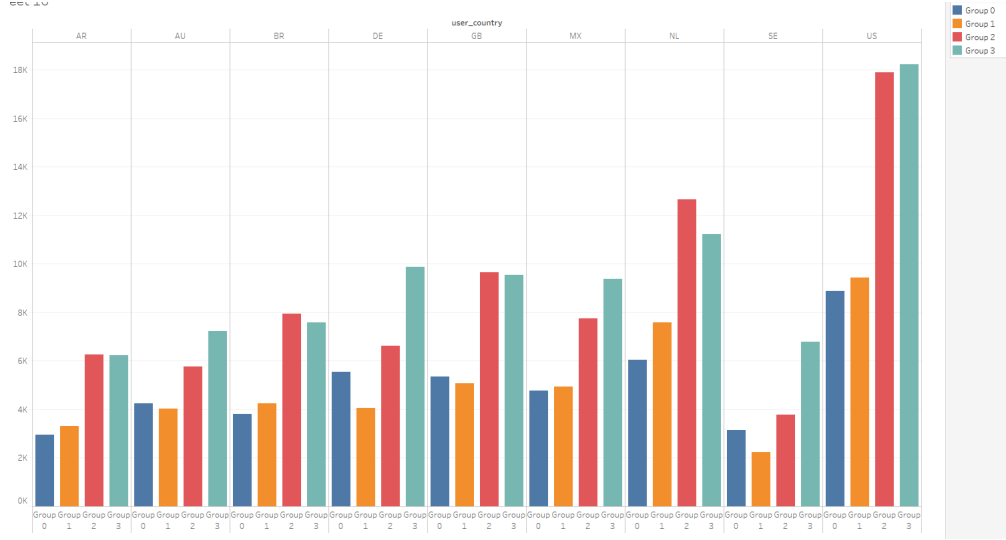
	Group 0	Group 1	Group 2	Group 3
Avg. danceability	0,5900	0,6475	0,6874	0,5393
Avg. energy	0,8392	0,5946	0,8088	0,8602
Avg. instrumentalness	0,7802	0,0384	0,0391	0,0341
Avg. valence	0,2627	0,3007	0,5772	0,2197

A couple key properties emerge from this table:

- Group 0 contains most of the instrumental tracks, hence Group 1, 2 and 3 contain most of the vocal tracks.
- Group 1 contains on average the least energetic tracks.
- Group 2 contains on average the happiest tracks.
- Danceability is not an identifying attribute.

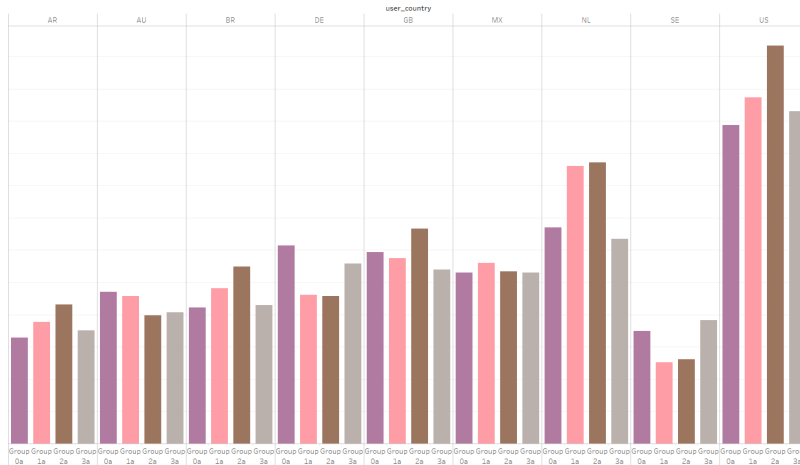
Now we compare these groups to other data.

Figure 3.8: Full data set: Class distribution for relevant countries



Judging from Figure 3.8, groups 2 and 3 are the most streamed groups. Group 2 also contains the two tracks with the highest number of records. There are a few differences between countries: Group 3 is a lot more commonly streamed in Sweden en Germany than group 2, for instance, whereas in the Netherlands group 2 has the highest number of records. However, we are currently not taking into account the support of each group, so these results are very absolute: Groups 2 and 3 have more tracks, so it makes sense for those groups to have more listens. If we divide the number of records for every group by its support, we obtain the weighted class distribution of Figure 3.9:

Figure 3.9: Full data set: Weighted class distribution for relevant countries



This gives us a different, more balanced distribution. Germany appears to be a country where instrumental tracks are more liked than elsewhere. People in the United States tend to listen more to happier tracks, since group 2 scores best there, which is the group with the highest valence. Interestingly, The Netherlands and Germany appear to be inversely related: While tracks from groups 1 and 2 are relatively more popular in The Netherlands, tracks from groups 0 and 3 do well in Germany. This kind of information can be used to market certain tracks to certain regions. For instance, if we classify a track as part of group 2, Armada can use this knowledge to campaign this track in The Netherlands, but maybe not in Germany.

Finally, we have another look at genders:

Table 3.5: Number of records per group per gender

	Group 0	Group 1	Group 2	Group 3
Male	647.237	786.996	1.735.102	1.619.891
Female	245.832	491.440	1.121.171	664.488
Female % (rounded)	28%	38%	39%	29%

From this table we can conclude that groups 1 and 2 are either more popular with women or less popular with men than groups 0 and 3, which indicates that the tracks women listen to more are located in groups 1 and 2, which points to high valence and low energy as characteristics of songs women prefer.

It is worth noting that we use streaming stats in most of the analysis, but not in the construction of our four groups. This gives the predictive potential of the groups more credibility.

3.4 Skiprate analysis

Using a track's duration, we can compute skip rates for every track. The skip rate of a track is the percent of unfinished plays or "skips". A finished play is defined by a certain threshold of listening, and anything below that is considered a skip. In our case, a full play is simply considered the full 100% of a track. It is worth noting that a stream is not registered in the platform until 30 seconds into the song.

This metric is useful for two reasons. Firstly, if a song has a very high skip rate, we can conclude that this song is not very successful with the general public and might not be worth advertising. More importantly, according to speculation within the business, a low skip rate gives a song a higher chance of appearing in a playlist curated by Spotify. Spotify has its own playlists based on various themes - think along the lines of "Summer Hits" or "Work-out Jams". These playlists are the most popular playlists on Spotify and generate a lot of streams and, therefore, revenue. Spotify tries to minimize costs by including songs in these playlists with low skip rates; for such songs they have to pay out less to artists and labels because streams last longer on average and so less tracks are being played. For Armada, getting a track in a Spotify-curated playlist would result in a large boost in streams and subsequently, revenue, which is ultimately what we're trying to accomplish. To compute the skip rate, we can determine which streams are partial plays by comparing the stream length variable to the full track duration. This gives us a column of partial plays and total plays. The skip rate is then defined by the following formula:

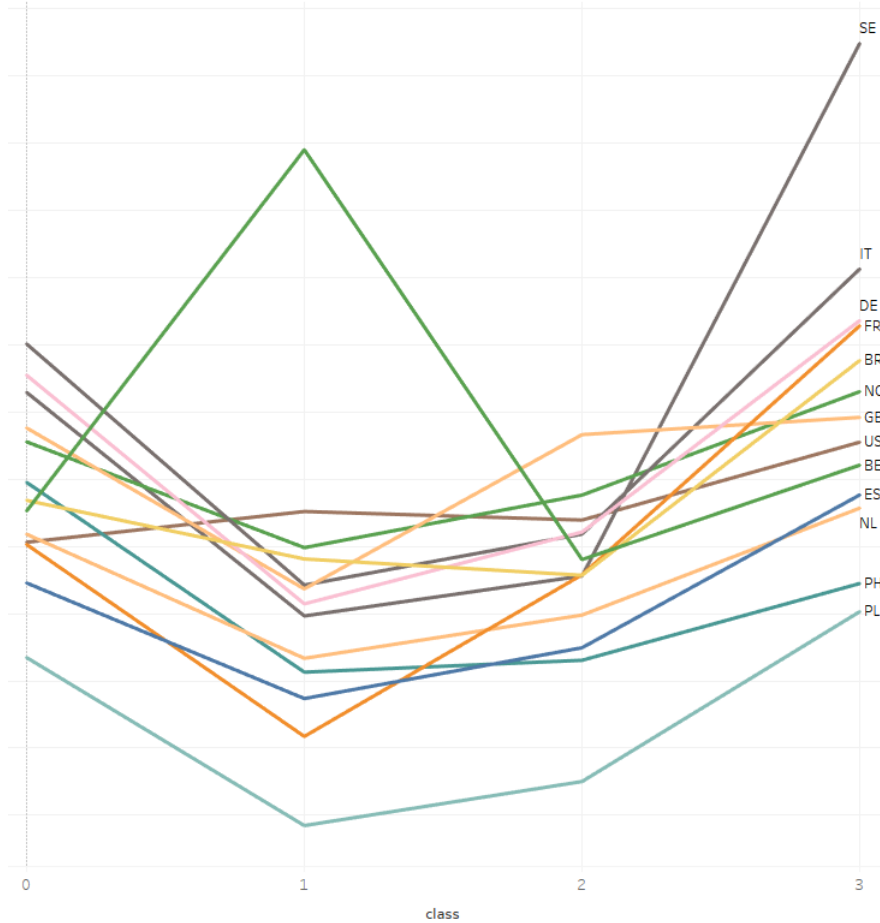
$$\text{skip rate} = \frac{\text{sum of partial plays}}{\text{sum of total plays}}$$

We can compute more measures using the track duration variable. For example, the play percentage is defined as follows:

$$\text{play percentage} = \frac{\text{average of stream length}}{\text{track duration}}$$

Using this measure, we look at the skip rates per group per country.

Figure 3.10: Average class skip rate per country



We draw a couple conclusions from this graph:

- On average, group 1 has the lowest skip rate. Belgium sticks out with a very high skip rate in group 1; looking at the track table, we can attribute this to a single track with a 75% skip rate and a relatively high stream count.
- Poland has the lowest skip rate, so advertising in this country could be very effective in getting tracks added to the Spotify curated playlists.
- Skip rates in group 0 and 3 are highest, with group 3 skip rates being significantly higher in Sweden than anywhere else.
- Groups 1 and 2 have on average the lowest skip rates, which coincides with female listeners preferring to listen to tracks in groups 1 and 2.

Chapter 4

Related Work

A large inspiration for this thesis was a paper written by Markus Schedl, "Investigating country-specific music preferences and music recommendation algorithms with the LFM-1b dataset". In it, Schedl uses a publicly available last.fm data set to determine differences in music preferences per country, based on genres. Using his metrics, Schedl implements a few recommendation algorithms and tests them. The idea to perform country-based analysis stems from this research, although our approach was quite different, since our methods of gathering data and the attributes we use for our research differ[6].

Along with Christine Bauer, Markus Schedl has also looked into mainstreamness in music, and how this can be used to recommend music to listeners. In their paper "Introducing Global and Regional Mainstreamness for Improving Personalized Music Recommendation", Bauer and Schedl propose several new mainstreamness measures and use these to build preference profiles for users and countries. Interesting observations in differences between countries are then explored, and finally these profiles are used to improve music recommendation[5].

In "Understanding user behavior in Spotify", Boxun Zhang, Gunnar Kreitz, Marcus Isaksson, Javier Ubillos, Guido Urdaneta, Johan A. Pouwelse and Dick Epema analyse behavioural patterns from Spotify users. They accomplish this by investigating system dynamics and individual user behaviour[3]. This paper has given us insights in useful measures to analyse, such as Spotify users' favourite times of day to listen to music.

Chapter 5

Conclusions

In conclusion, we have demonstrated a number of analytical options for this data set. We have shown that it is capable of yielding usable results.

More specifically, we have demonstrated the following:

- We have shown that we can plot birth year data in different ways, resulting in an accessible overview. Using a treemap gave us a more detailed overview while retaining a clear view, making it the preferable choice for displaying birth years.
- We have identified differences in listening behaviour between male and female people. Certain tracks perform better in certain groups. Gendered campaigns could help these tracks to obtain a larger audience.
- We can discover a daily listening pattern. Music is mostly listened to during the afternoon, with peak hours occurring during common commute hours. Campaigning for tracks might be more effective during these hours.
- There exists a clear difference in people's listening sources on mobile compared to on desktop. Mobile users prefer their own collection of music, desktop users more often listen to other people's playlists. Campaigns on apps that are mostly aimed towards mobile users are best applied to tracks that are often saved to people's own playlists. Similarly, desktop campaigns work best for new discoveries.

We have also developed a classification system to group tracks based on properties, and introduced a skip rate measure. Using these new findings, we have identified more properties:

- Certain track properties perform better in certain countries. This can be used to start campaigns for tracks in only a certain number of countries, based on its classification.

- Skip rates differ not only per country, but also per group within every country. Since low skip rates are beneficial to Armada, it can be useful to look into which tracks are skipped less and put more effort into campaigns for those tracks.
- The male-to-female skip rate ratios are different for certain groups of tracks with specific properties. Female listeners appear to prefer tracks with lower energy and higher valence.

Bibliography

- [1] Nancy Chinchor. Muc-4 evaluation metrics. *Proc. of the Fourth Message Understanding Conference*, page 22–29, 1992.
- [2] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.
- [3] B. Zhang et al. Symmetric lenses. In *2013 Proceedings IEEE INFOCOM*, pages 220–224. IEEE, 2013.
- [4] J MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1:281–297, 1967.
- [5] Christine Bauer Markus Schedl. Introducing global and regional mainstreamness for improving personalized music recommendation. In *Proceedings of the 15th International Conference on Advances in Mobile Computing Multimedia*, pages 74–81. ACM New York, NY, USA, 2017.
- [6] Markus Schedl. Investigating country-specific music preferences and music recommendation algorithms with the lfm-1b dataset. *International Journal of Multimedia Information Retrieval*, 6:71–84, 2017.
- [7] Ben Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics (TOG)*, 11:92–99, 1992.