Business requirement

Development unsupervised Machine Learning Moosic Model

Client: Moosic GmBH Wien, Austria

Prepared by Damir Selak

Content

- Business requirement, product results and activities description
- Algorithm methodology
- Playlists presentation
    - Example of 1 song from a few different clusters.
- PROs and CONs of using clustering to create playlists
    - K-Means method advantages and disadvantages to create playlists
    - Recommendation for moving forward with K-Means algorithm
    - Recommendation for moving forward with other methods to create playlists **(not considered)**
- PRESENT DATA SET FROM KMEANS
    - Are Spotify’s audio features able to identify “similar songs”, as defined by humanly detectable criteria?
- Additional data for better user experience (production years)
- Recommendation and next steps
- Learnings from the project

Business requirement, product results and activities description

Business requirement:

Development unsupervised Machine Learning Moosic Model (MLM) by “Automatisation of Moosic playlists”

Results:

R1. Algorithm for unsupervised Machine learning Model (MLM) developed
R2. Algorithm successful tested and functionality confirmed

Business requirement, product results and activities description

Main activities:

A1. Importing and reviewing data frame quality

A2. Cleaning and scaling data frame

A3. Plotting dataframe in Histogram to visually check correlation between several pairs of chosen data values to be used for K-Clustering

A4. Plotting data frame in scatter plot to visually check correlation between several pairs of chosen data values to be used for K-Clustering

A5. Choosing the right number of cluster, using inertia

A6. Bringing decision on optimum number of clusters and set of data pairs which will be used for algorithm development

A7. Data scaling Quantile transformer

A8. K-means calculation for chosen pairs of data for clustering x="energy", y="acousticness"

A9. Plotting and exploring our KMeans results / Comparing our centroids and our dataset

A10. Adding new cluster column to our data frame and defining the names of 10 clusters based on music genre

A11. Data frame analysis to check cluster structure

A12. Visualizing the clusters in a scatterplot with K-means features x="energy", y="acousticness"

A13. Explore relation between sum of energy and accusticness

A14. Ploting relation between energy and accusticness

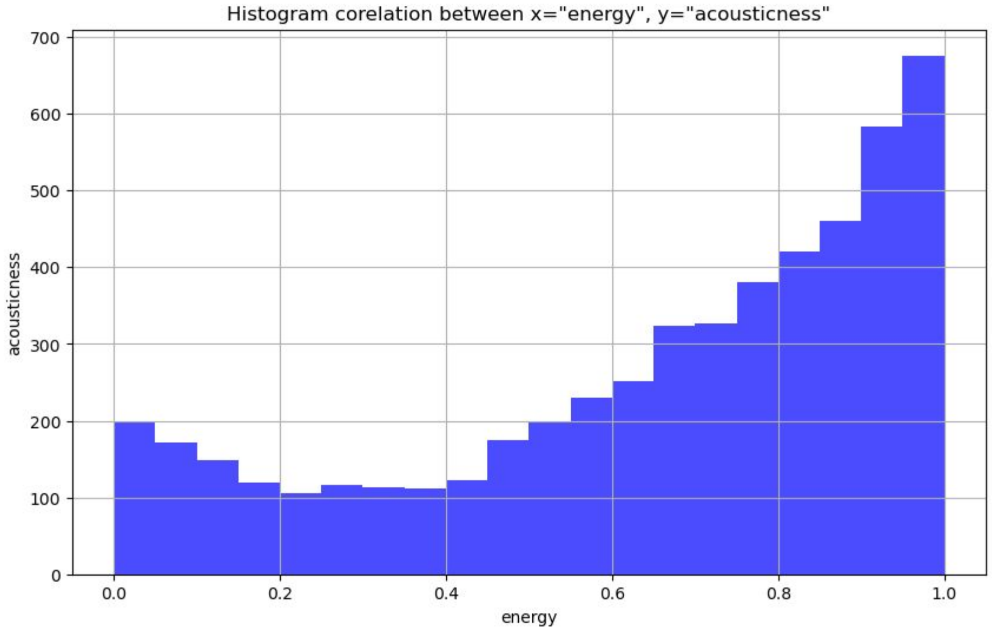
Algorithm methodology

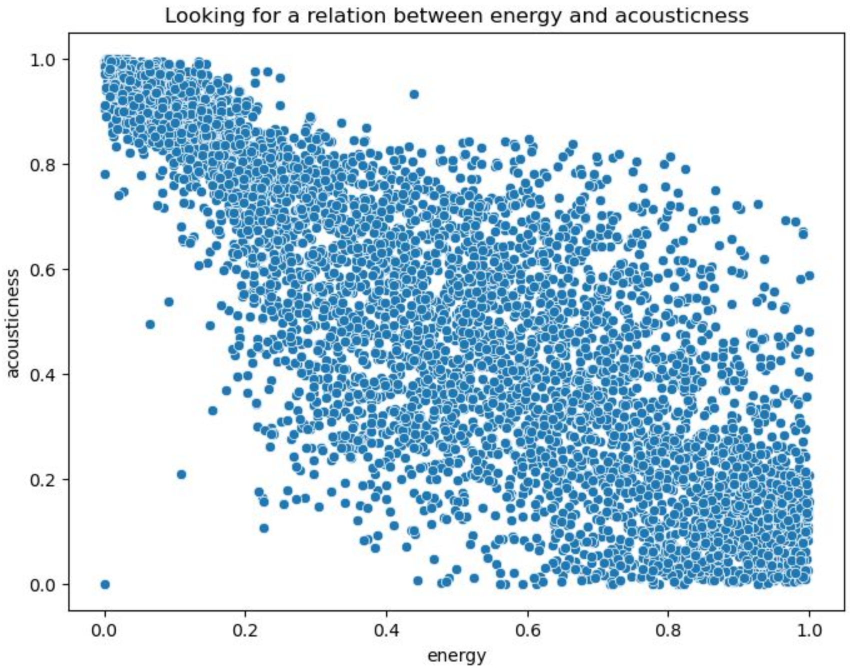
- **First we explored meaning of the different data** set values in order to understand music meaning of the i.e. accusticness, energy, valence, loudness, etc
- After that we **plotted in scatter and histogram plot several pairs of chosen data values to visually check correlation** between them
- Choosing **x="energy", y="acousticness"** as **basic values for clusterisation as those two data values showed significant correlation** increase of the energy and decrease of the accusticness in histogram and descending function of same values in scatter plot. ***DATA SET IS NICELY STRETCHED*** **next slide**

- Using **Inertia** to find the right number of clusters **"10"** and The **K-means algorithm** for clustering, **next slide…**

- Data set cleaning and scaling using robust preprocessing scheme **Quantile transformer**
  - Advantages:
    - transforms the features to follow a uniform or a normal distribution
    - tends to spread out the most frequent values
    - reduces the impact of (marginal) outliers
    - Recommended for ML models
  - Disadvantages:
    - this transform is non-linear.
    - It may distort linear correlations

Algorithm methodology Chosing x="energy", y="acousticness"





Algorithm methodology k-means pro and cons

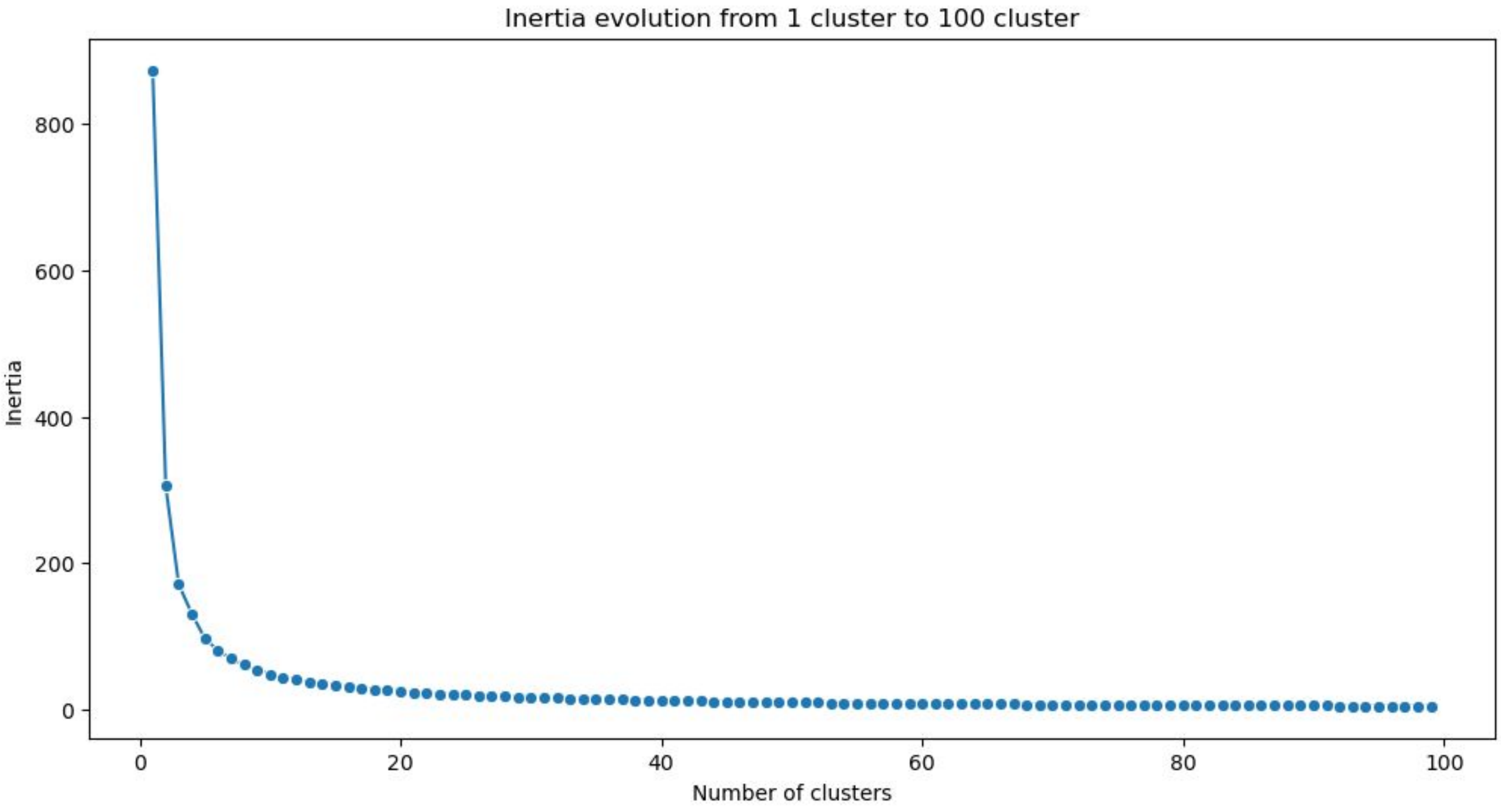
Advantages of k-means algorithm

- Relatively simple to implement.
- Scales to large data sets.
- Guarantees convergence (ALL DATA ON ONE PLACE)
- Can warm-start the positions of centroids.
- Easily adapts to new examples.
- Generalizes to clusters of different shapes and sizes, such as elliptical clusters.

Disadvantages of k-means

- Being dependent on initial values.
- Clustering data of varying sizes and density.
- Clustering outliers Centroids can be dragged by outliers, or outliers might get their own cluster instead of being ignored . Minimized using Quantile transformer
- Scaling with number of dimensions

Algorithm methodology INERTIA



Playlists presentation

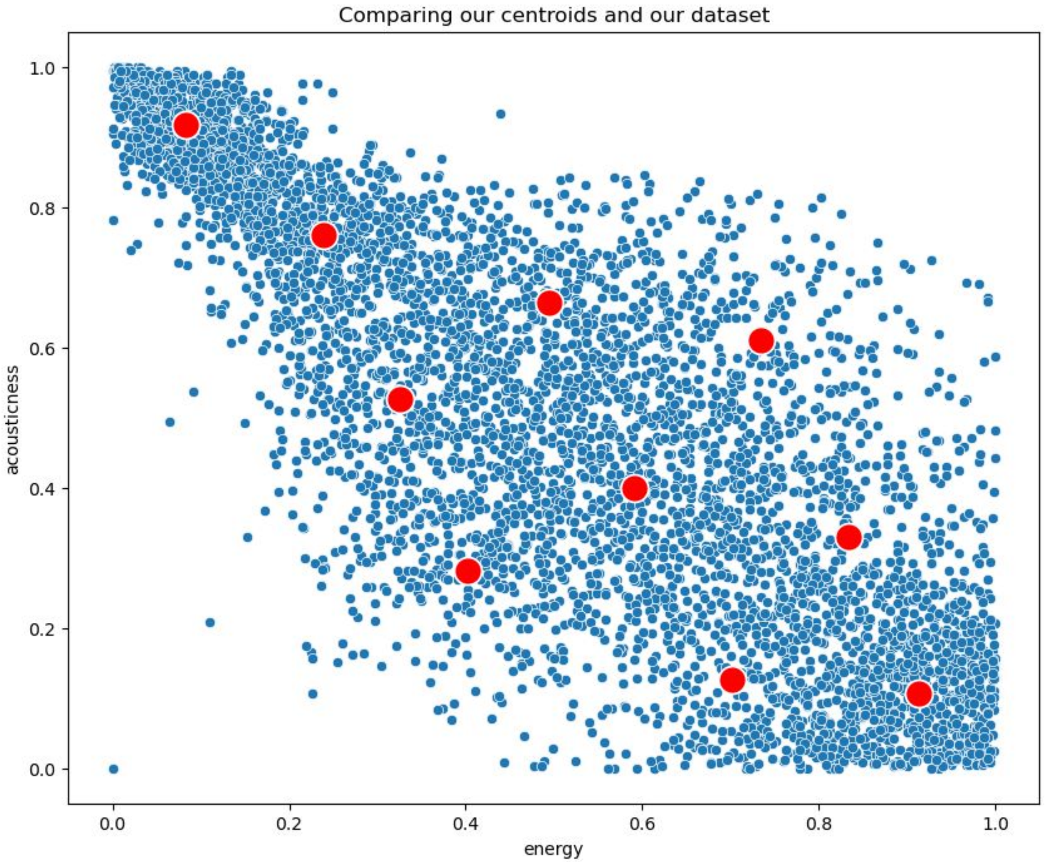
- Example of 1 song from a few different clusters.

artist	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	type	duration_ms	cluster
Gilberto Gil	0.658	0.259	11	-13.141	0	0.0705	0.694	0.000059	0.975	0.306	110.376	blouse	256213	1
Antônio Carlos Jobim	0.742	0.399	2	-12.646	1	0.0346	0.217					soul	191867	6
Martinho Da Vila	0.851	0.73	2	-11.048	1	0.347	0.453					rock	152267	3
Chico César	0.705	0.0502	4	-18.115	1	0.0471	0.879					rap	186227	4
Kurt Elling	0.651	0.119	6	-19.807	1	0.038	0.916					rap	273680	4

type	artist	cluster
blouse	658	1
classic	377	5
jazz	548	2
modern	708	0
new age	404	8
pop	432	7
rap	830	4
relax	324	9
rock	483	3
soul	471	6
Sum	5235	10

Observe "energy" and "acusticness" as it correlate negatively

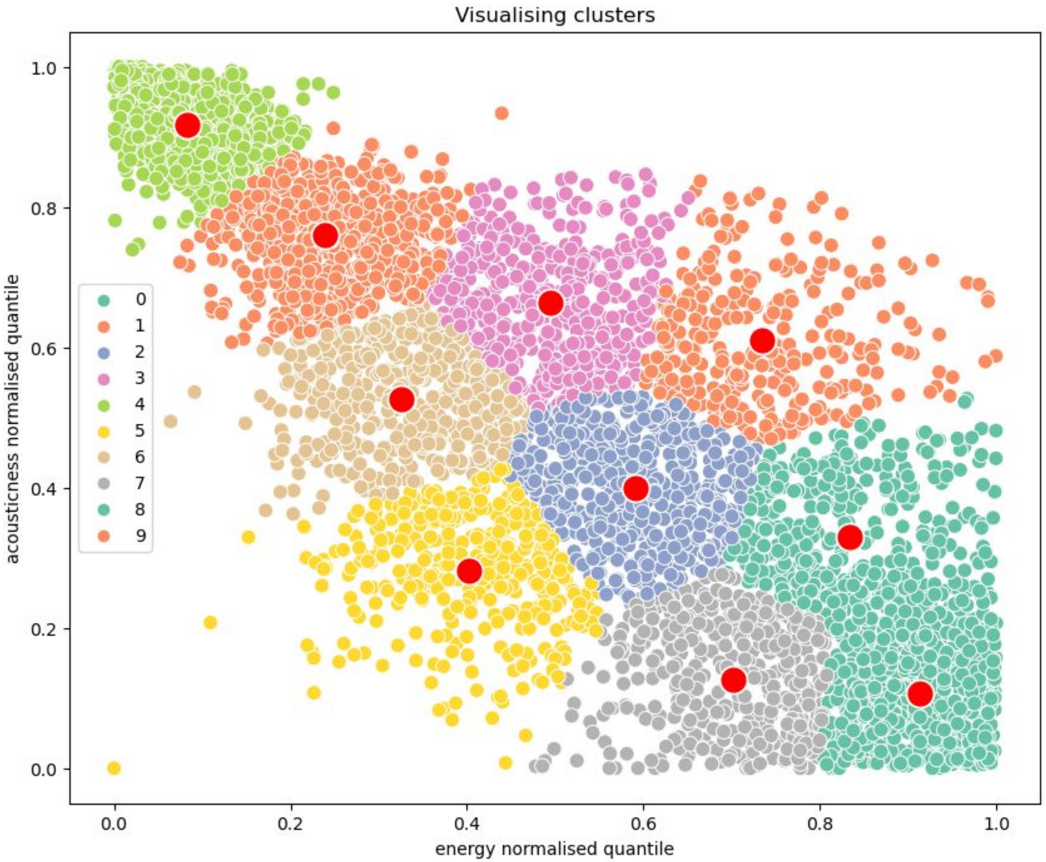
PRESENT DATA SET FROM KMEANS



Index	original	quantile_transformed
3643	0	0
3414	0	0
3453	0	0
3849	0	0
3928	0	0
...	...	...
2097	0.996	1
2099	0.996	1
1928	0.996	1
1974	0.996	1
4566	0.996	1
5235 rows × 2 columns		

Transformation or aspect ratio using quantile transformer

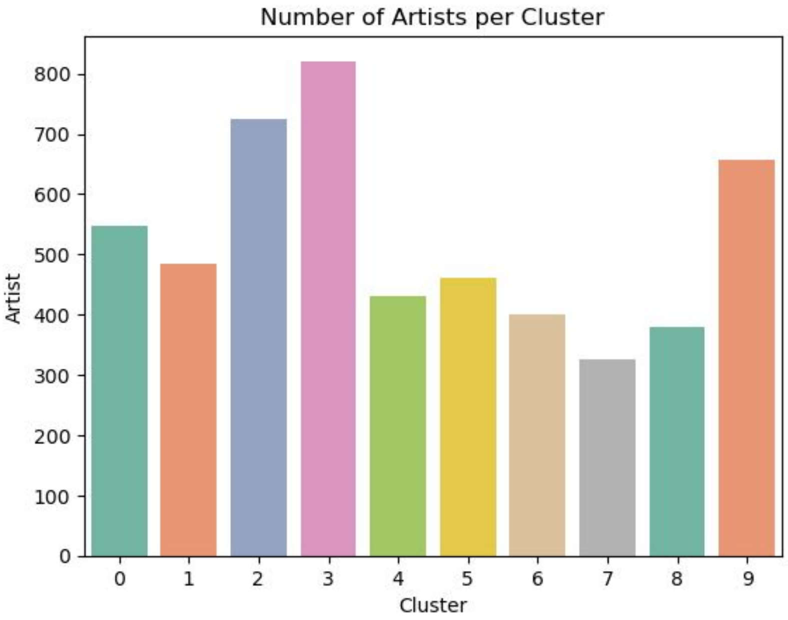
PRESENT DATA SET FROM KMEANS



type	artist	cluster
blouse	658	1
classic	377	5
jazz	548	2
modern	708	0
new age	404	8
pop	432	7
rap	830	4
relax	324	9
rock	483	3
soul	471	6
Sum	5235	10

Data frame and clusters analysis

Explore and Plot number of artists per clusters based on genre classification to check structure of the clusters and potentially propose importing new melodies to client



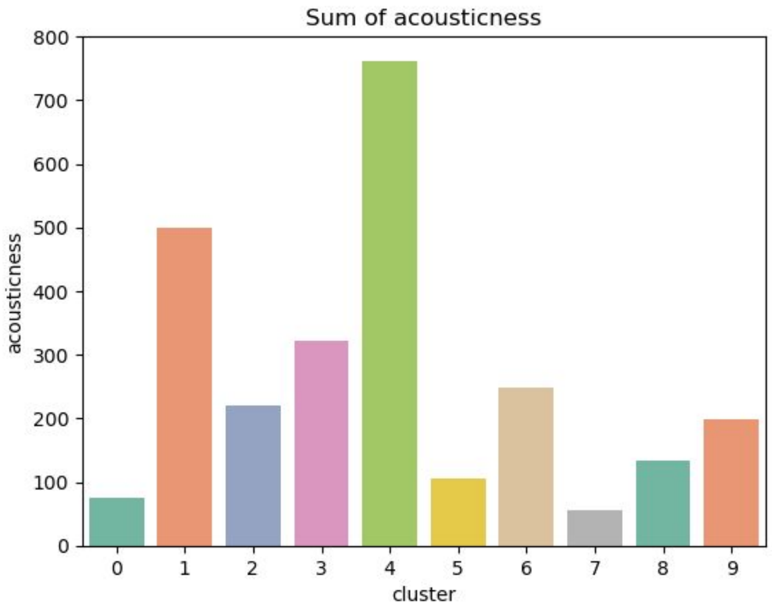
Additional melodies and artist should be included in the playing list clusters with lower number of artist and melodies i.e.

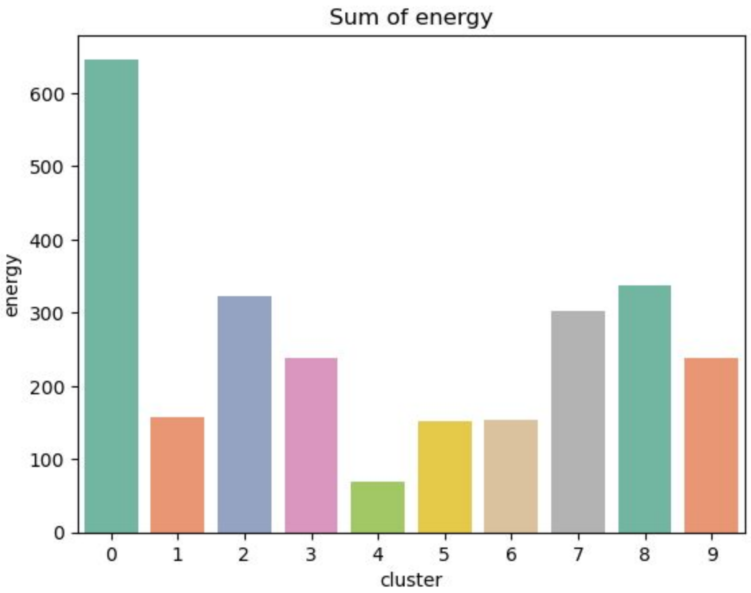
 # soul = 6

 # pop = 7

Data frame and clusters analysis

EXPLORE and plot RELATION BETWEEN SUM OF ENERGY AND ACUSTICNESS





It generaly accepted opinion that **today pop and generally modern music is less acoustic and more energetic** than in the 1950s [Pop music is louder, less acoustic and more energetic than in the 1950s | Digital music and audio | The Guardian]

Our clustering decision is confirmed by this wide accepted opinion, but in order to prove it our data frame should be amended with additional column "Song_year_created"

Data frame and clusters analysis

EXPLORE and plot RELATION BETWEEN SUM OF ENERGY AND ACUSTICNESS

Pop music is louder, less acoustic and more energetic than in the 1950s

But its danceability hasn't changed from Elvis Presley to Miley Cyrus, according to music tech firm The Echo Nest

Modern music is just noise. You can't hear the words properly. Those electronic things aren't proper instruments. Why is it all so loud? You can't dance to this, not like in my day.

Data alchemist" Glenn McDonald, running tests on the 5,000 hotttest [sic] tracks from 1950 to 2013 to see how specific attributes – including energy, loudness, organicness, acousticness and mechanism – **have changed over that time.**

TIME IS IMPORTANT FACTOR in MUSIC!

Are Spotify's audio features able to identify "similar songs", as defined by humanly detectable criteria?

- Spotify Criteria are
  - Relevant
  - well structured and defined
  - **BUT, tempo and instrumentalness** are maybe **over & under estimated** that **could impact on results of melodies clusterization (greater number of outliers)**

artist	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	type
Gilberto Gil	0.658	0.2590	11	-13.141	0	0.0705	0.694	0.000059	0.975	0.306	110.376	

- **Average person** most likely **will not be capable** to distinct and search melodies based on the spotify classification criteria as those criteria are complex for understanding

Therefore our recommendation is as follow on next slides…

Additional data for better user experience

- As premises for our algorithm methodology and analysis is based on **melodies energy and accusticness** which are generaly connected to music creation period (year), we believe that including **year of the melodies production** can bring better user experience

- Additional melodies and artist should be included in the playing list clusters with lower number of artist and melodies i.e.

  # soul = 6

  # pop = 7

Recommendations & Learnings from the project

Recommendations

Classification of the users experience based on "Song_year_created" which proved to be connected with “energy and accusticness” is solid basis for creating Unsupervised Moosic Maschine Learing Model that will automate classification of music based on proposed “genre” structure.

Learnings from the project

- Based on data frame **first decision** should be which ML model to chose
- It is important to recognise **how set of data correlate** in order to choose correct data pairs for exploration
- **ML algorithms prefer certain models** and scaling methodologies than oders, **explore this before**
-
- I finally resolve issue from “sniping tool” and better learned Anaconda JupiterLab