Spotify Content Based Recommender System

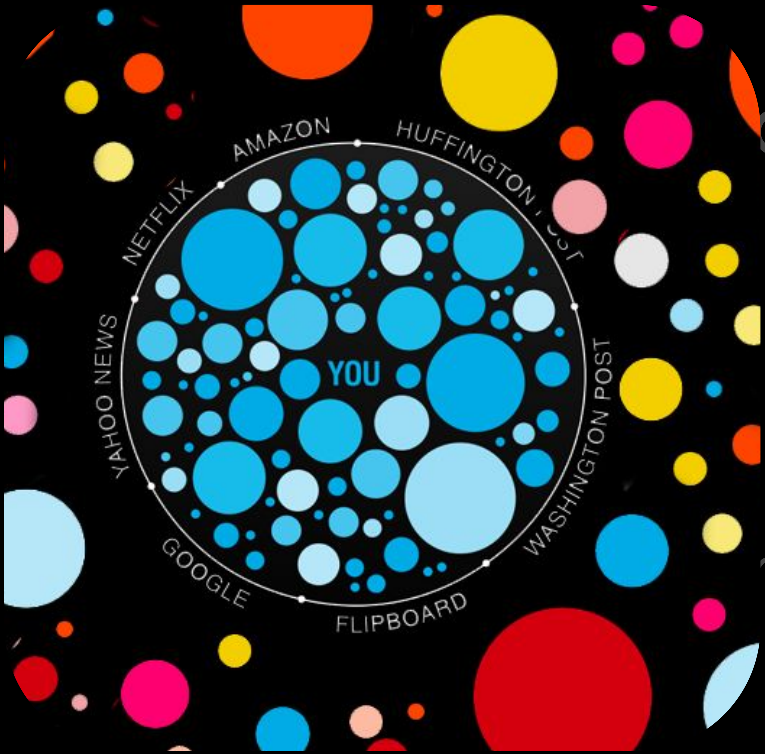
Preventing the echo chamber effect

What is an echo chamber?

Environment where a person only encounters information or opinions that reflect their own.

- Recommendations based on user data
- Algorithm trains on data
- Diversity in recommendations shrink over time

Economically: Lowered product demand
Socially: Divided and divisive society



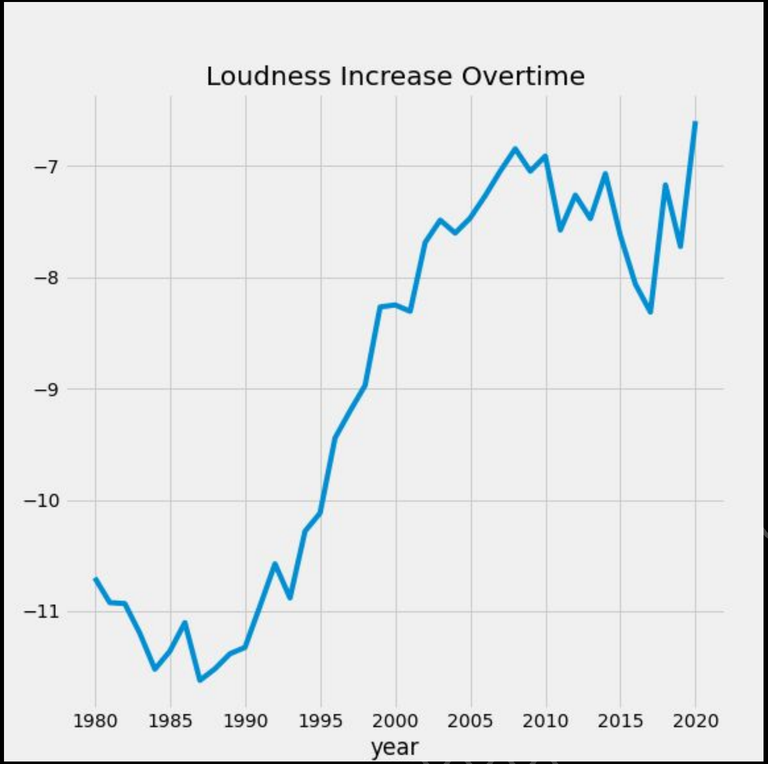
Why content based?

- Spotify uses collaborative filtering for recommendations
  - Based on similar users
  - Most users listen to popular music
- Recommendations will be based on music attributes (Spotify Kaggle Dataset)

Music Trends

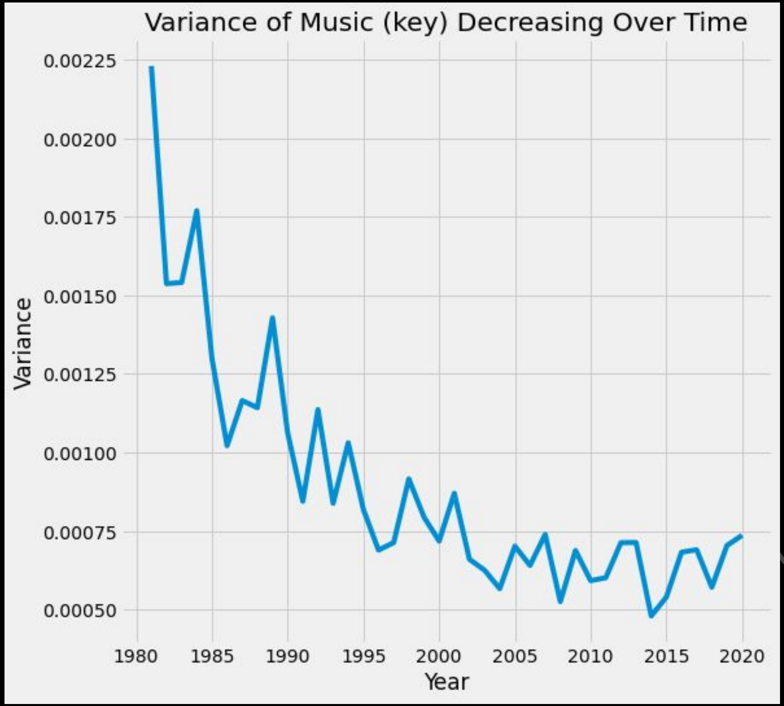
Loudness

- Inherent volume of music before adjustment by listener
- Louder tracks tend to grab listener's attention
- Reduced the dynamic range of the music



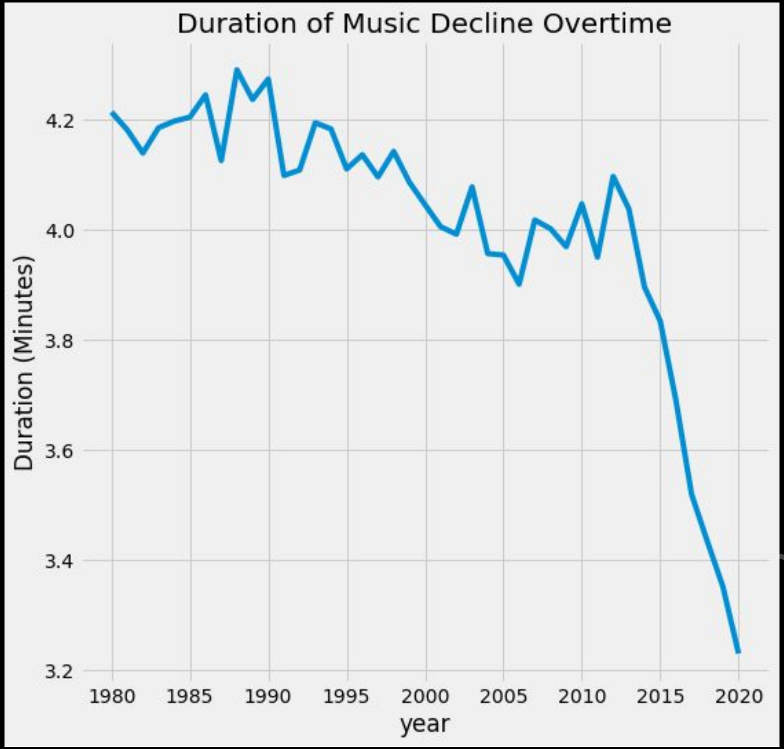
Variance in Key

- Key included details about harmony, melody, chords, and progressions
- Variety of songs in different keys has shrunk over the years
- Musicians are becoming less inventive and adventurous



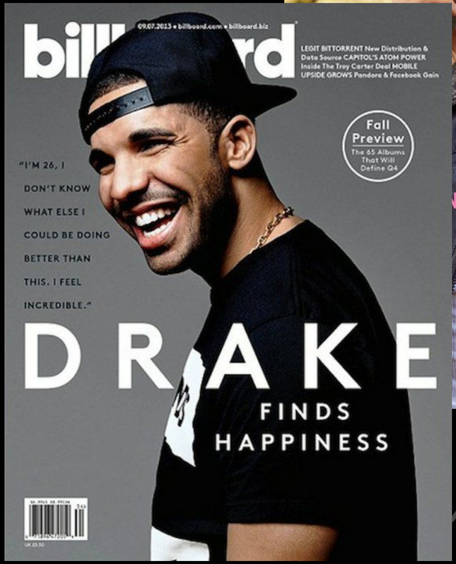
Music Duration

- Gradual decrease until 2011
- Sharp decline after
- Spotify will only pay after a certain amount of a song is played



How Spotify is changing the way we consume music

- The ‘Spotify Sound’
- tracks to feature the hook, guest artist or a prominent sample in the first few seconds
- Chart music become more homogenized

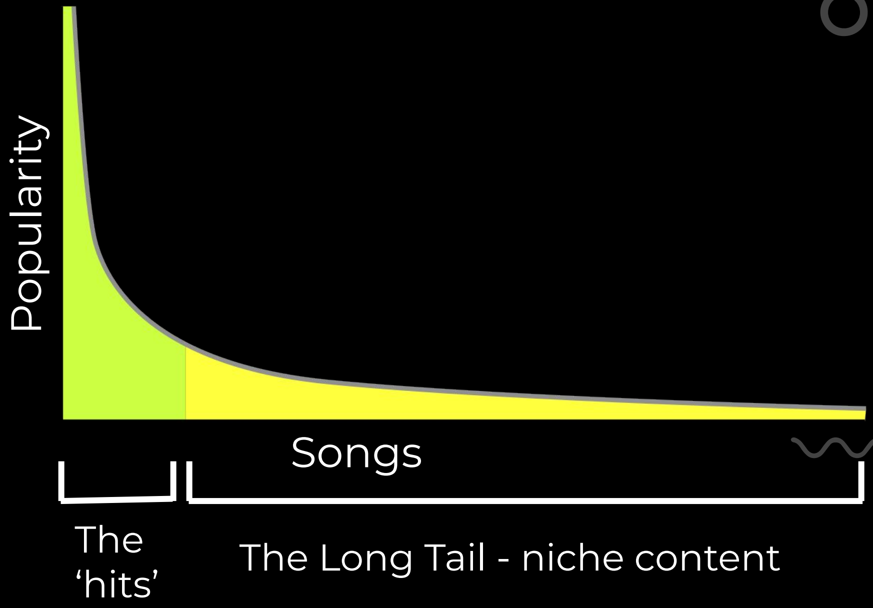


What is the goal?

- Encourage listeners to broaden their horizon
  - Recommend songs of different genre, eras and languages

- Increase demand in different types of music

- Prevent the homogenization of music



Metrics

Diversity

- ○ Are different genres recommended?
- ○ Are different languages recommended?
- ○ Are songs from different eras recommend?

- If any of question above answer is yes, Diversity metric satisfied.

Novelty

- ○ How new, original, or unusual the recommendations are

- Low Novelty Score = songs that have been heard before or popular
- High Novelty Score = Songs that have never been heard before or niche

Building the Recommender

Song Attributes

Danceability

Energy

Speechiness

Loudness

Mode

Instrumentalness

Tempo

Liveness

Valence

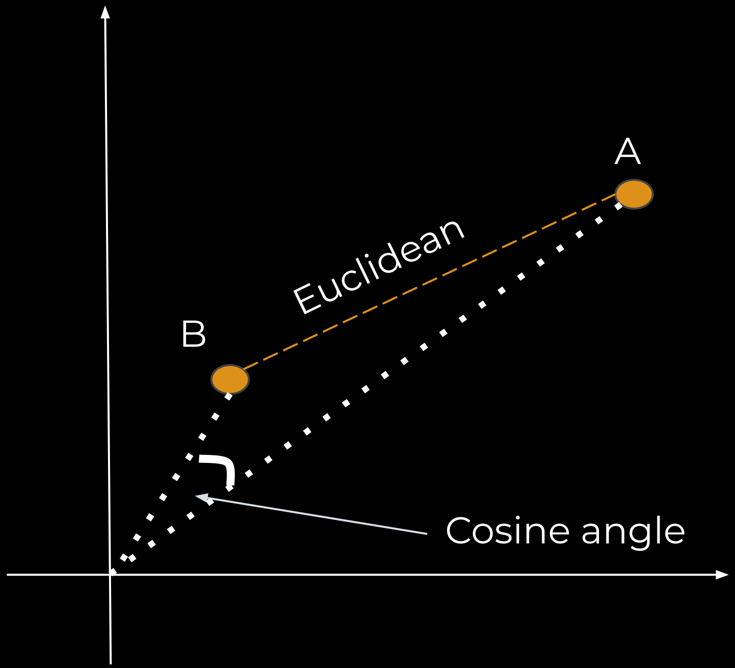
The Dataset

- 170K songs after cleaning
- Dropped features:
  - Duration_ms
  - Popularity
  - Year
  - Explicit
  - Release_date

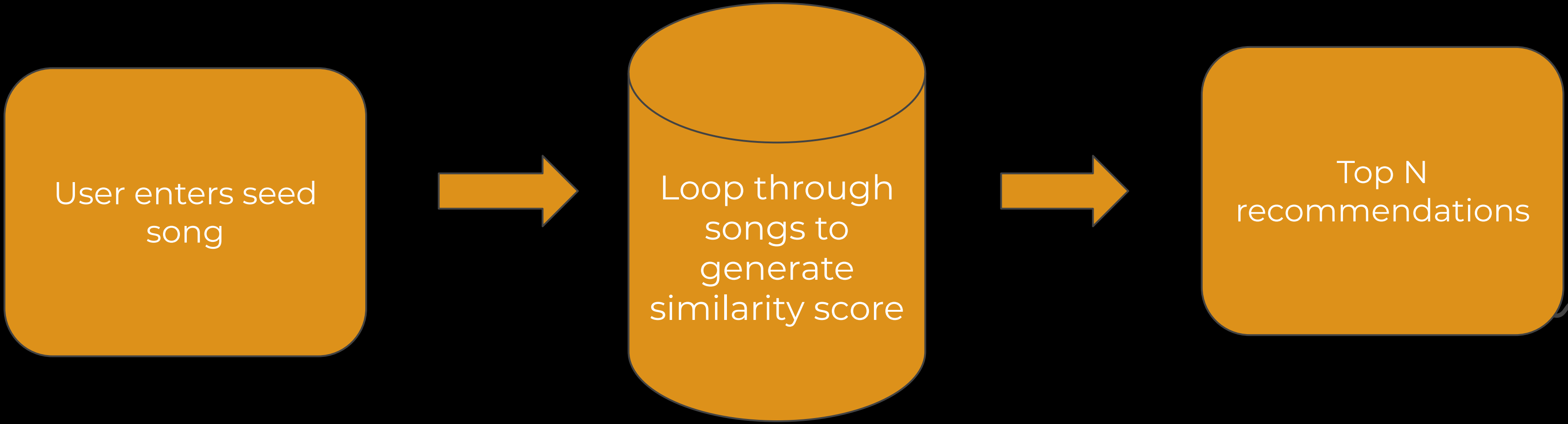
Using Cosine Similarity for Top-N

- Advantage over euclidean distance:
    - Even though, euclidean distance might be large between points.

    - their angle might be small, and therefore high similarity.



How the recommender works (Top - N)



2 Songs used to test the Recommender

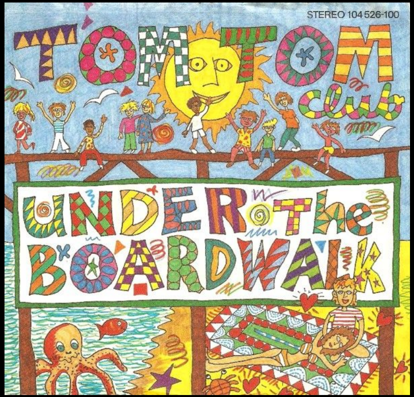
Vanilla Ice - Ice Ice Baby
90s Hip-Hop

John Legend - All of Me
2010s Pop/Ballad.

Ice Ice Baby - Vanilla Ice

Top 4 songs recommended

- Epochs: From 1970 -1992
- Genres: funk, rock, hip hop, Electronic
- Languages: English only



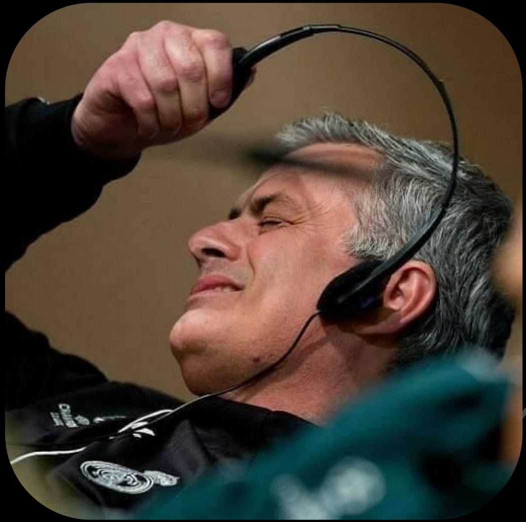
All of Me – John Legend

Top 4 songs recommended

- Epochs: From 1949 -2017
- Genres: Ballads, Folk
- Languages: Arabic, French English, Spanish

Evaluating the Recommender

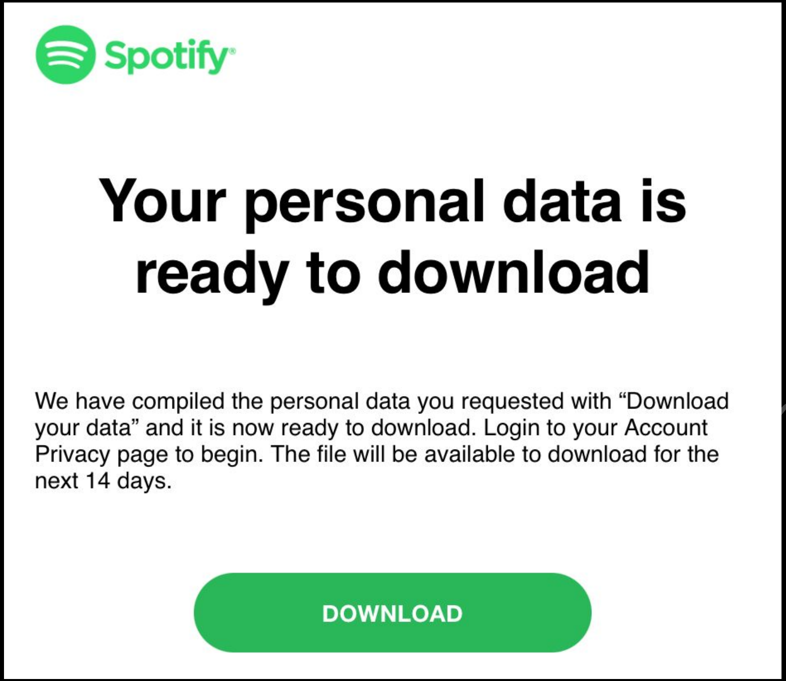
- Satisfy the Diversity
- May appear to be a random recommender from a random one
  - Most songs were Novel
- Decrease novelty to make recommender more trustworthy



Personalizing the Recommendations

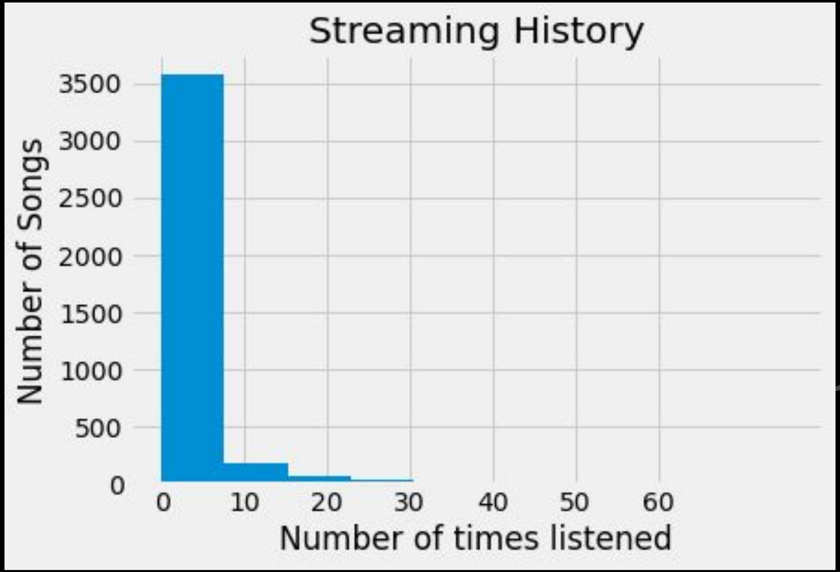
Performing Classification on my own Data

- Comes in json format
- Includes streaming history , separated by listening sessions per song
- Listening duration per song
- 9k streamed songs (without music attributes)
- Use to classify songs I like



Determining what is a potential favourite

- Clear drop off from 0-10 listens
- Assigning songs with >= 10 listens as 1
- Assigning songs with < 10 listens as 0
- Training set



Modelling and Comparing Precision

- Random Forest, ADA Boost, XGBoost
- Baseline = 93.2% Accuracy
- Using SMOTE to handle imbalanced class
- Precision as the deciding factor

Model Selection

	Accuracy	Precision
Baseline	93.2%	-
Random Forest	93.4%	91.4%
Ada Boost	89.6%	92.9%
XGBoost	94.6%	95.1%

Personalized Dataset

Kaggle Dataset

170k songs

XGBoost

(predict potential favourite, class 1)

Kaggle Dataset

Filtered to only have class 1.

24,269 songs

+

Streaming History (training set)

Filtered to only have class 1.

262 songs

=

Personalized Dataset

Only has existing and potential favourites.

24,531 songs

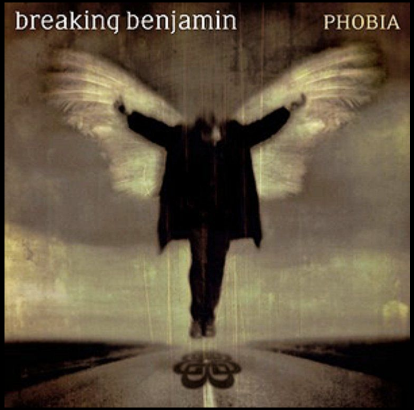
Ice Ice Baby - Vanilla Ice

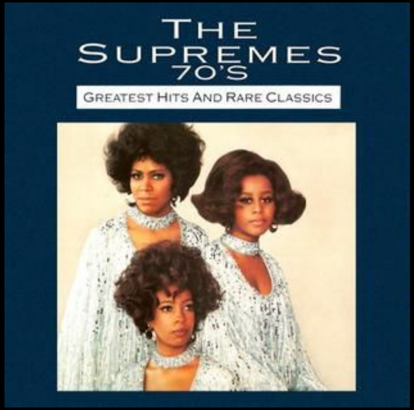
songs recommended:

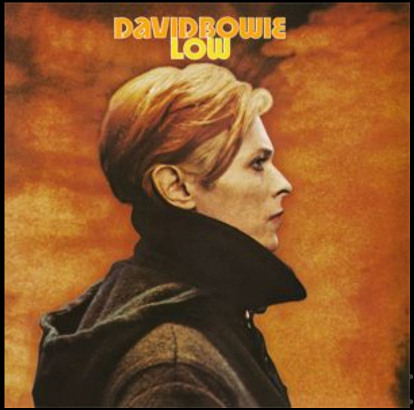
Epochs: From 1964 -2009

Genres: r&b, rock, hip hop, **glampop**

Languages: English and **Spanish**







All of Me - John Legend

songs recommended:

Epochs: From 1993 -2017

Genres: **Hip Hop**, **R&B**

Languages: **English only**

Evaluating the Personalized Recommender

- Lower Novel score
  - Songs were familiar
- Lower Diversity
  - Only hip hop for John Legend song
- Reduced coverage
  - Trade off for predicting potential favourites with diversity

Improving the Recommender

Adding Weights for Recency Bias

	Liveliness	Energy	Acousticness	...	Weights
Song 1	0.9	0.1	0.8	...	**0.3**
Song 2	0.5	0.7	0.6	...	**0.7**
Song 3	0.8	0.6	0.1	...	**0.85**
Song 4	0.4	0.4	0.1	...	**1**

Final Playlist Vector	0.445	0.324	0.561	...

This vector is used to compare to other song vectors

Would it work?

Gathering Feedback

19 Responses total

Questions (Scale 1 - 5, 3 = Neutral)	Mean Score
• How similar do the songs in the survey playlist sound to you?	3.4
• Do you feel that Spotify recommends you music that is outside your comfort zone (i.e preferred genre/artist) ?	2.6
• As an added feature, would you like if Spotify recommended similar songs based only on their audio features?	4.1

Open ended feedback

“ It's also **refreshing** to see music from different eras introduced so that users are able to expand their music library.”

“I think when listening to the playlist casually, I couldn't spot any similarities. **Only when i listened carefully**, did I pick up on the technical similarities”

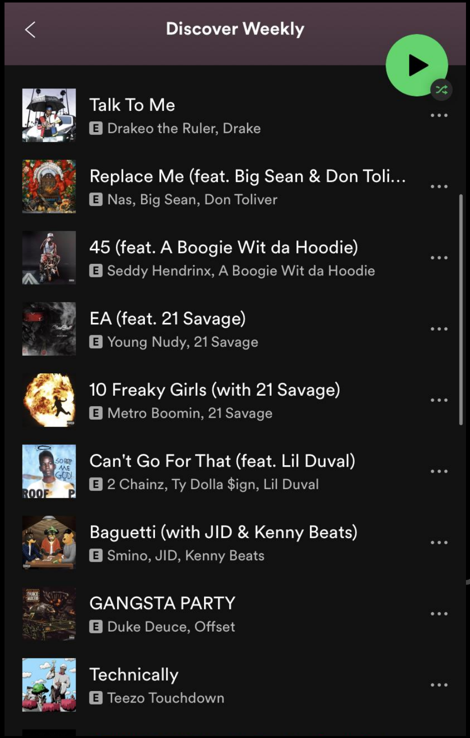
“It’s **interesting** to hear the similar groove in different songs but it **doesn’t flow as well**”

“Personally, **i build playlists based on mood** so tempo, volume, unique sounds/patterns and key are the prominent features.”

“Eclectic but all a similar tempo and speed.”

Conclusion

- The recommender generates diverse recommendations
  - addresses a problem that is not common knowledge
- Better for casual listener
- Marketed as a personalized radio
- Not replacing Spotify's current recommender, but an added feature
- Product differentiation strategy



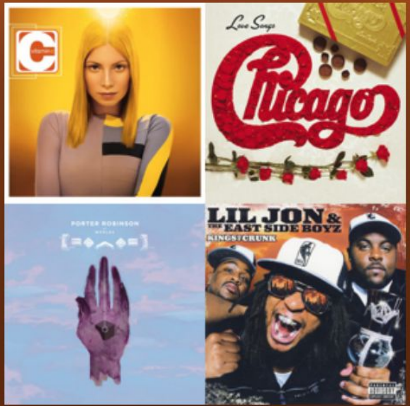
Limitations

- Personalization depends on User streaming history
- Unsupervised learning
- Playlist tend to be created based on moods

Recommendations

- Work with lyrics
- A/B Testing
    - Engagement metrics

Sounds Like Graduation!



PLAYLIST

Sounds Like Graduation!

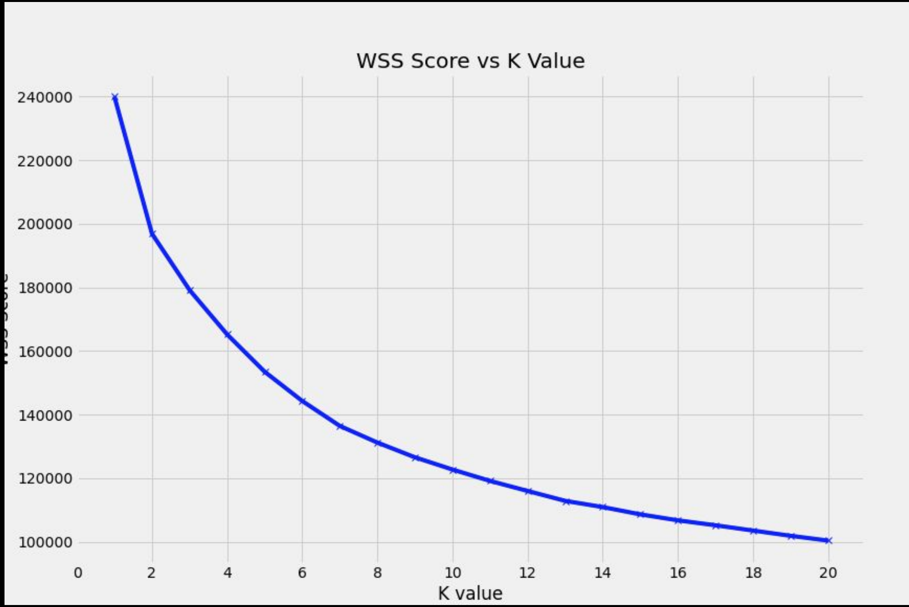
Adel Alsagoff • 8 songs, 40 min 28 sec

#	TITLE	ALBUM
1	Graduation (Friends Forever) Vitamin C	Vitamin C
2	Will You Still Love Me? - 2003 R... Chicago	Love Songs
3	Divinity Porter Robinson, Amy Millan	Worlds
4	Get Low E Ying Yang Twins, Lil Jon & The Eas...	Kings Of Crunk
5	Gloria En Lo Alto Christine D'Clario	De Vuelta Al Jardin
6	Pretty Boy M2M	Shades of Purple
7	Dreamer Ozzy Osbourne	Down To Earth
8	Stigmatized The Calling	Camino Palmero


Thank You For Your Attention!

KMeans and Elbow Method

- Plotting K against WSS
- Within-Cluster-Sum of Squared Errors = Compactness of clustering
- Choose K where adding another K does not improve Score by much.
- Optimal K = 8



Where does Bach fall under?

- Tendency to cluster around Cluster 1
- Some songs can be found in other cluster
- Illustrating genres cannot be clustered based on audio features
- We can expect different genres in different cluster