



Aziz Maredia | GA Capstone | 1.27.2021

Agenda

- Project & Web Application Overview
- Data Collection
- Data Cleaning
- Principal Component Analysis
- Clustering Model
- Recommender System
- Application Demo
- Conclusions & Future Outlook



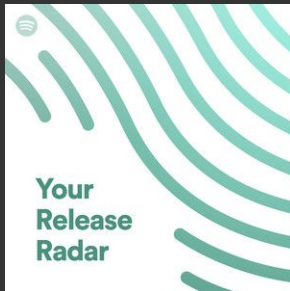
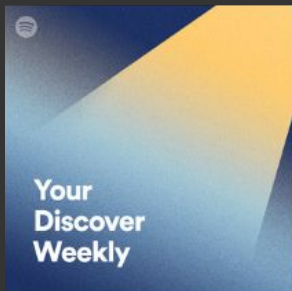
Web Application Overview



Hate spending time discovering new songs you love?

Sortify ranks an album's songs based on your listening history so you can find the music you love, faster.

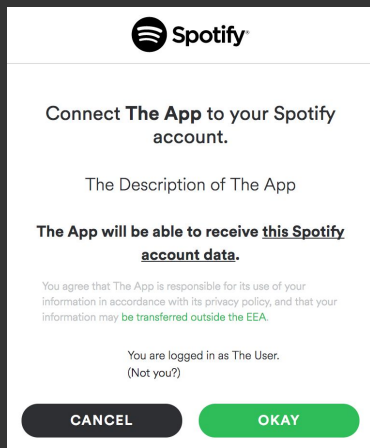
Any album, sorted to match your taste.



Data Collection

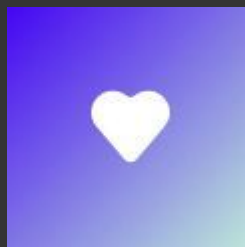


1. User grants permission for app to access data

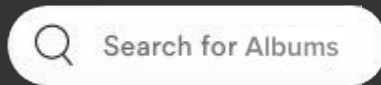


/authorize

2. Pull user's saved library & tracks of album searched



/v1/me/tracks



/v1/search → /v1/albums/{id}

3. Pull features using track and artist IDs



/v1/audio-features /v1/artists



Data Cleaning

- Bring certain variables down to a 0-1 scale for clustering
 - Manually → Loudness
 - Min/Max Scaler → Tempo
- Convert certain categorical values into dummy variables
 - Key, Time Signature, Release Date
- Reduce variation of genre names
 - More info on next slide



Data Cleaning → Genres

Original Data
Dummified



track	ohio rap	alt hip hop	nyc rap	big room
# 1	1	1	0	0
# 2	0	0	1	1

Track 1 → ['ohio rap', 'alt hip hop']

Track 2 → ['nyc rap', 'big room']

Cleaned Data
Dummified



track	rap	hip hop
# 1	1	1
# 2	1	0

Track 1 → ['rap', 'hip hop']

Track 2 → ['rap']



Principal Component Analysis

Original Data → 40+ features

track	tempo	genre indie	key a	90s music	etc.
# 1	0.45	1	0	1	...
# 2	0.13	0	1	0	...
# 3	0.27	0	1	1	...

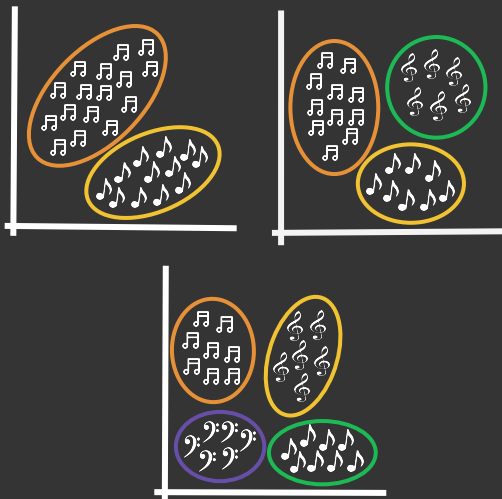
New Data → Dimensionality Reduction

	track	PCA 1	PCA 2	PCA 3	
→	# 1	0.45	1	...	←
→	# 2	0.13	0	...	←
	# 3	0.27	1	...	



K-Means Clustering

1. Test different values of K from 3 through 12



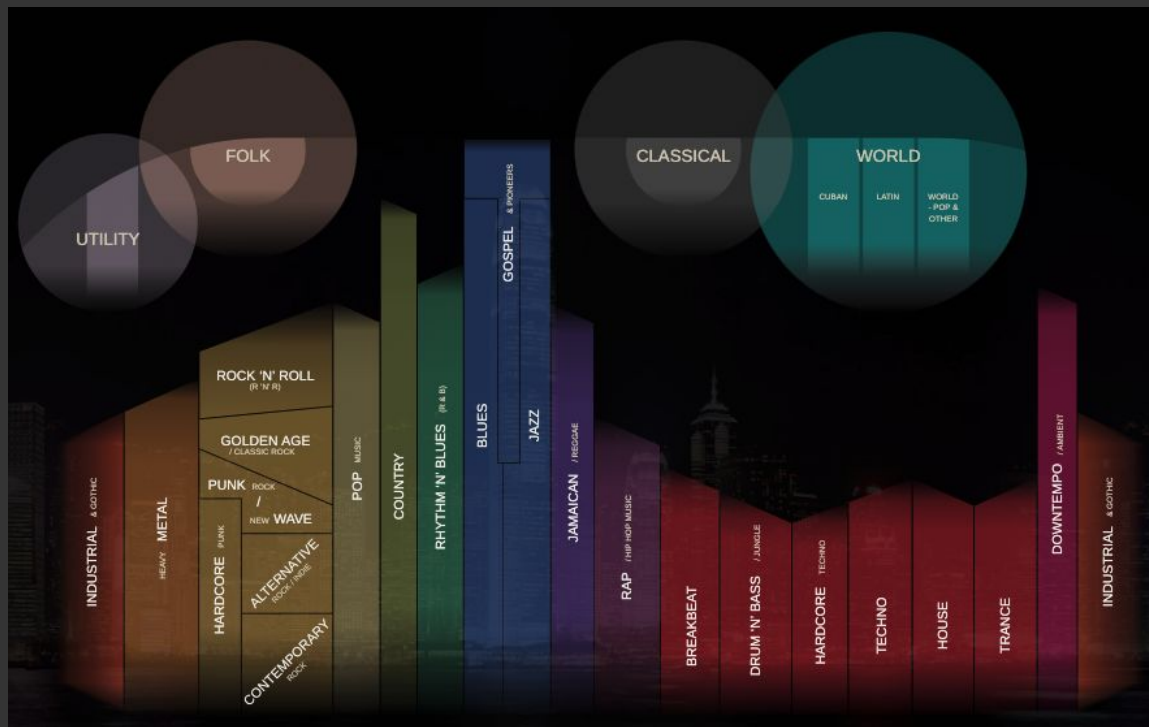
2. Choose cluster with highest Silhouette Score



3. Find coordinates of cluster centers







Music Map Used to Determine k-Value Test Range




Recommender System

1. Calculate cosine similarity between cluster centers and album tracks

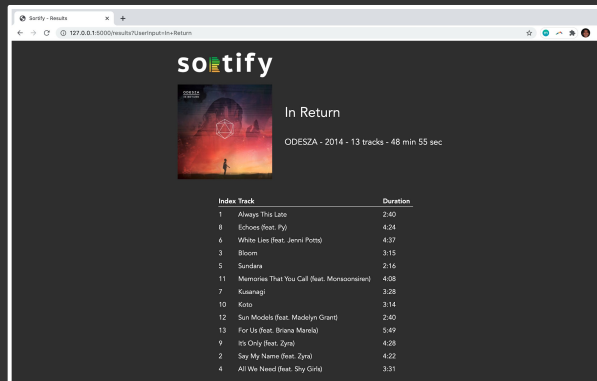
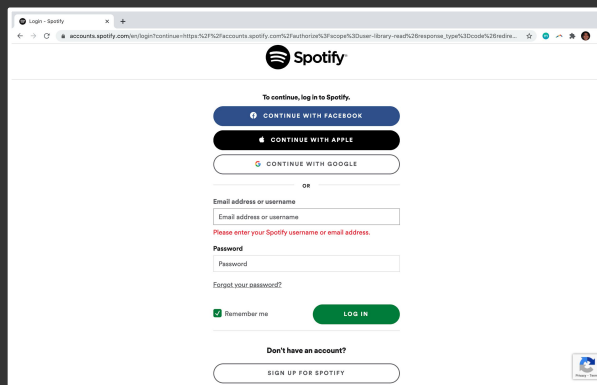
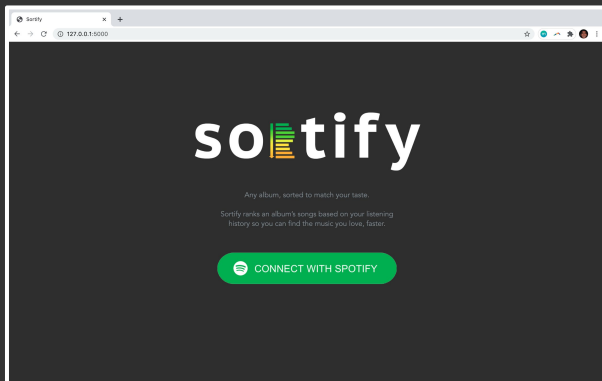
Track				
# 1	0.56	0.23	0.82	0.26
# 2	0.13	0.67	0.56	0.14
# 3	-0.34	0.05	0.79	0.38
# 4	0.63	-0.44	0.91	0.47
Mean	0.25	0.13	0.77	0.32

2. Sort album using scores from cluster with highest average cosine similarity

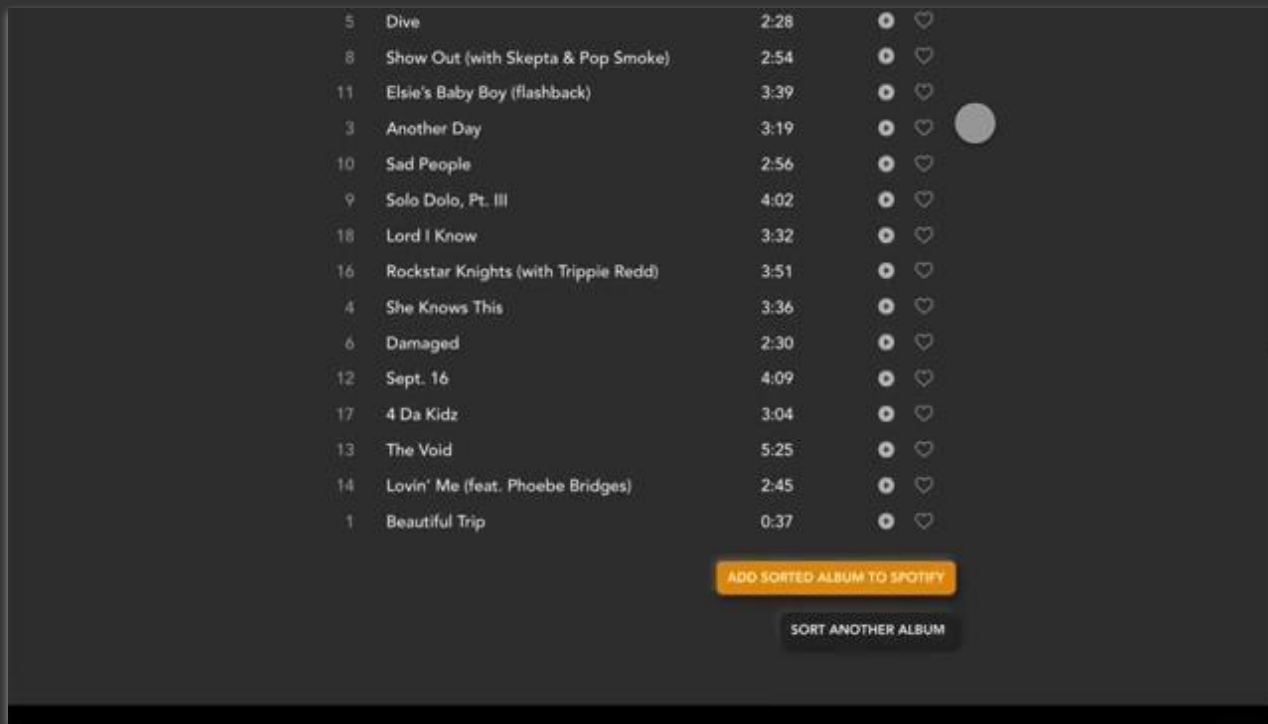
Track	
# 4	0.91
# 1	0.82
# 3	0.79
# 2	0.56



Web Application



Future Development - Prototype v2



Conclusions & Future Outlook

1. Develop web application v2 → February 2021
2. Look out for API updates to see if new features can be collected
3. Add lyrics and sentiment analysis to potentially improve recommendations
4. Try to optimize PCA components and k-value range to improve recommendations → Conduct user tests / surveys
5. Find ways to track songs that users dislike to help build a more robust user profile





Thank You!

Questions?