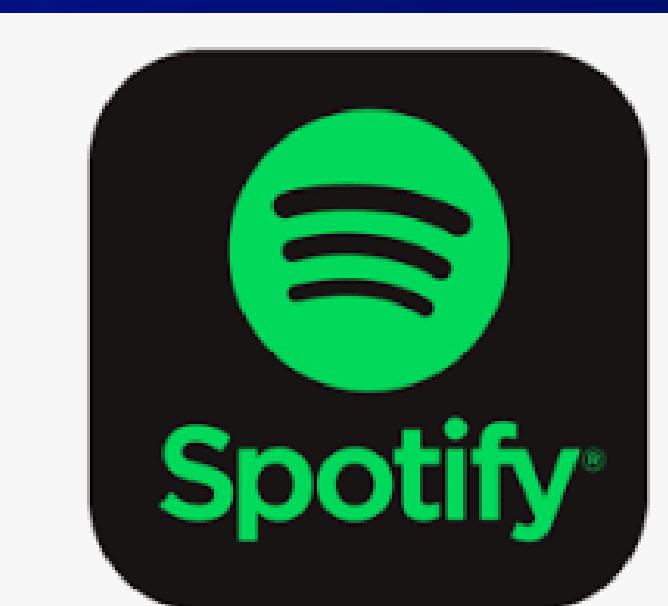


Spotify Tracks DB

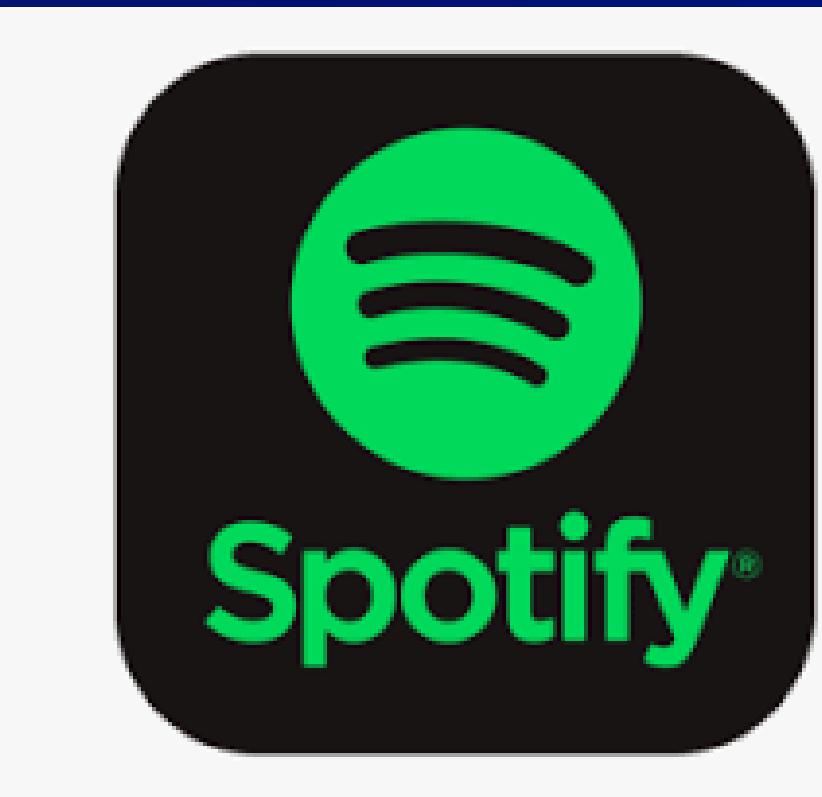
Data visualization



lets meet the data set

The dataset contains information about tracks (music) on Spotify

Accurate to the year 2021
Tabular data set



lets meet the data set

The dataset has 232,725 tracks (rows)

Each track has 18 feature (columns)

dtypes:

float64(9)

int64(2)

object(7)

memory usage: 32.0+ MB

No Null values

first 5 tracks:							
5 rows x 18 columns pd.DataFrame							
genre	artist_name	track_name	track_id	popularity	acousticness	danceabilit	tempo
Movie	Henri Salvador	C'est beau de faire un Show	0BRj06ga9RKCKjfDqeFgWV	0	0.611	0	140.0
Movie	Martin & les fées	Perdu d'avance (par Gad Elmaleh)	0BjC1NfoE00usryehmNudP	1	0.246	0	120.0
Movie	Joseph Williams	Don't Let Me Be Lonely Tonight	0CoSDzoNIKCRs124s9uTVy	3	0.952	0	140.0
Movie	Henri Salvador	Dis-moi Monsieur Gordon Cooper	0Gc6TVm52BwZD07Ki6tIvf	0	0.703	0	140.0
Movie	Fabien Nataf	Ouverture	0IuslXpMROHdEPvSL1fTQK	4	0.950	0	140.0

#	Column	Non-Null Count	Dtype
0	genre	232725 non-null	object
1	artist_name	232725 non-null	object
2	track_name	232724 non-null	object
3	track_id	232725 non-null	object
4	popularity	232725 non-null	int64
5	acousticness	232725 non-null	float64
6	danceability	232725 non-null	float64
7	duration_ms	232725 non-null	int64
8	energy	232725 non-null	float64
9	instrumentalness	232725 non-null	float64
10	key	232725 non-null	object
11	liveness	232725 non-null	float64
12	loudness	232725 non-null	float64
13	mode	232725 non-null	object
14	speechiness	232725 non-null	float64
15	tempo	232725 non-null	float64
16	time_signature	232725 non-null	object
17	valence	232725 non-null	float64

Values distribution

The feature distribution is not normalized.

The scales of the values are different (negative, large, small and ranges)

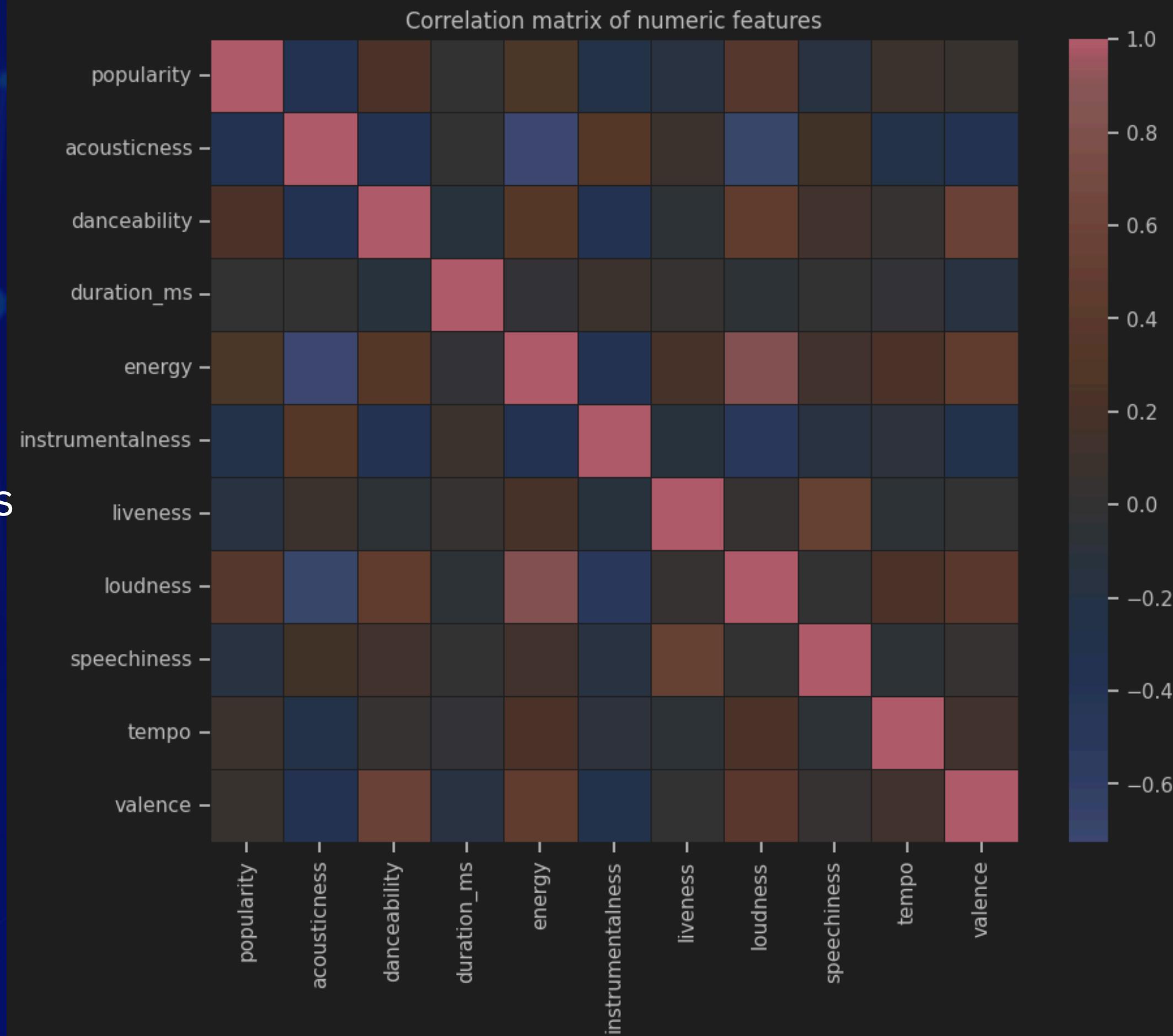
For some, the mean is not close to the median (good if some genres are far)

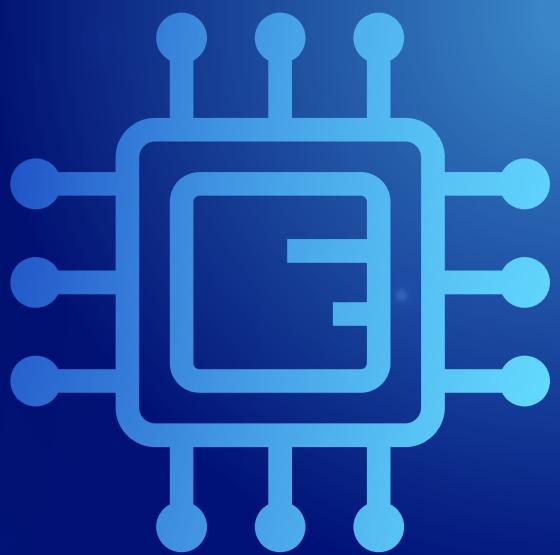
	count	mean	std	min	25%	50%	75%	max
popularity	232725.0	41.127502	18.189948	0.00000	29.0000	43.000000	55.0000	100.000
acousticness	232725.0	0.368560	0.354768	0.00000	0.0376	0.232000	0.7220	0.996
danceability	232725.0	0.554364	0.185608	0.05690	0.4350	0.571000	0.6920	0.989
duration_ms	232725.0	235122.339306	118935.909299	15387.00000	182857.00000	220427.000000	265768.0000	5552917.000
energy	232725.0	0.570958	0.263456	0.00002	0.3850	0.605000	0.7870	0.999
instrumentalness	232725.0	0.148301	0.302768	0.00000	0.0000	0.000044	0.0358	0.999
liveness	232725.0	0.215009	0.198273	0.00967	0.0974	0.128000	0.2640	1.000
loudness	232725.0	-9.569885	5.998204	-52.45700	-11.7710	-7.762000	-5.5010	3.744
speechiness	232725.0	0.120765	0.185518	0.02220	0.0367	0.050100	0.1050	0.967
tempo	232725.0	117.666585	30.898907	30.37900	92.9590	115.778000	139.0540	242.903
valence	232725.0	0.454917	0.260065	0.00000	0.2370	0.444000	0.6600	1.000

Feature correlation

From my perspective, it's safe to say the correlation between our features is not significant.

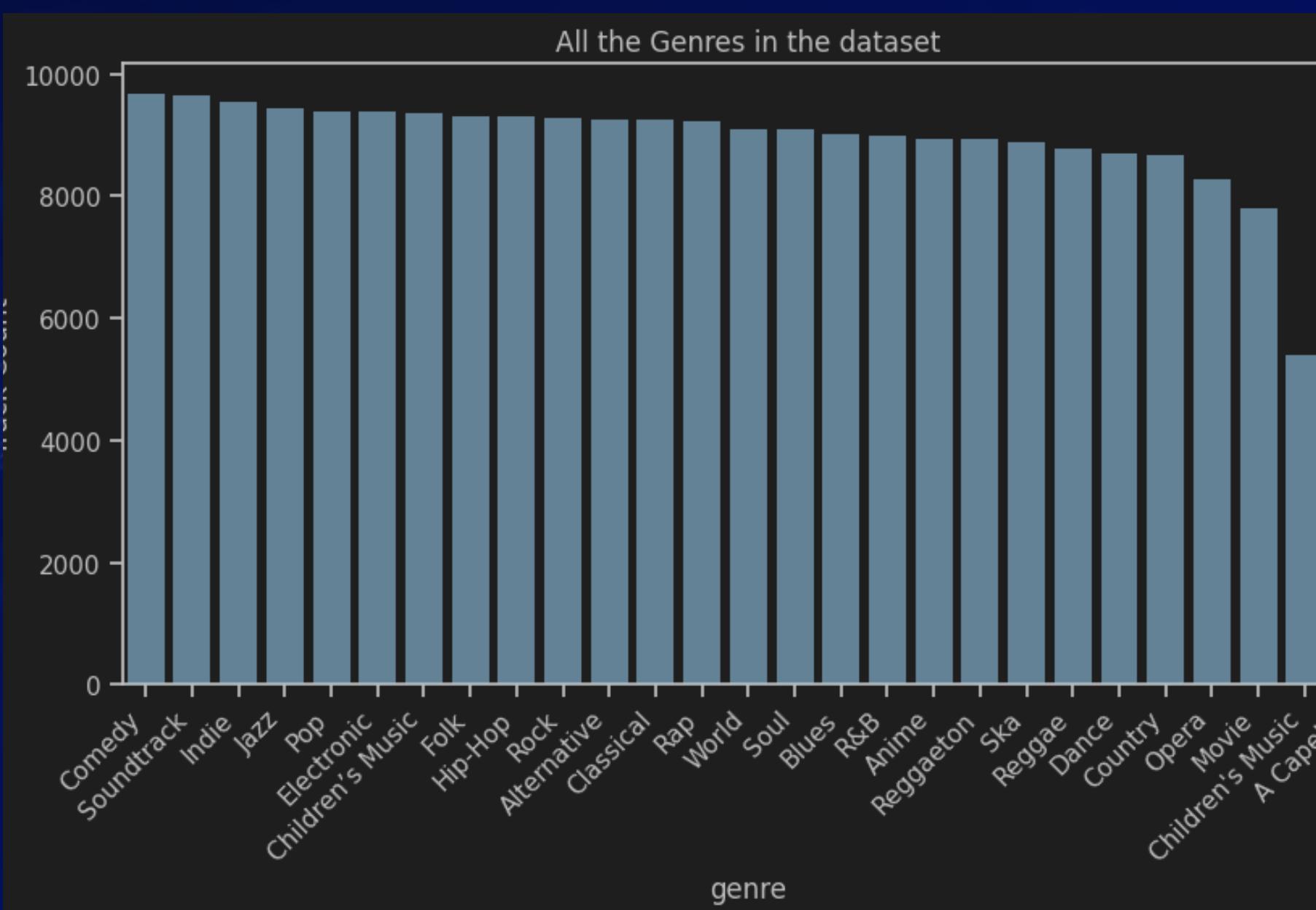
Which means that no feature is unnecessary because it can be represented by another.





Lets talk genres!

We got 27 genres
Each has up to 10,000 tracks
It contains very few aCapella tracks



I exported a CSV that contains the mean, median, and std of each value in each genre

It seems like an interesting thing to plot

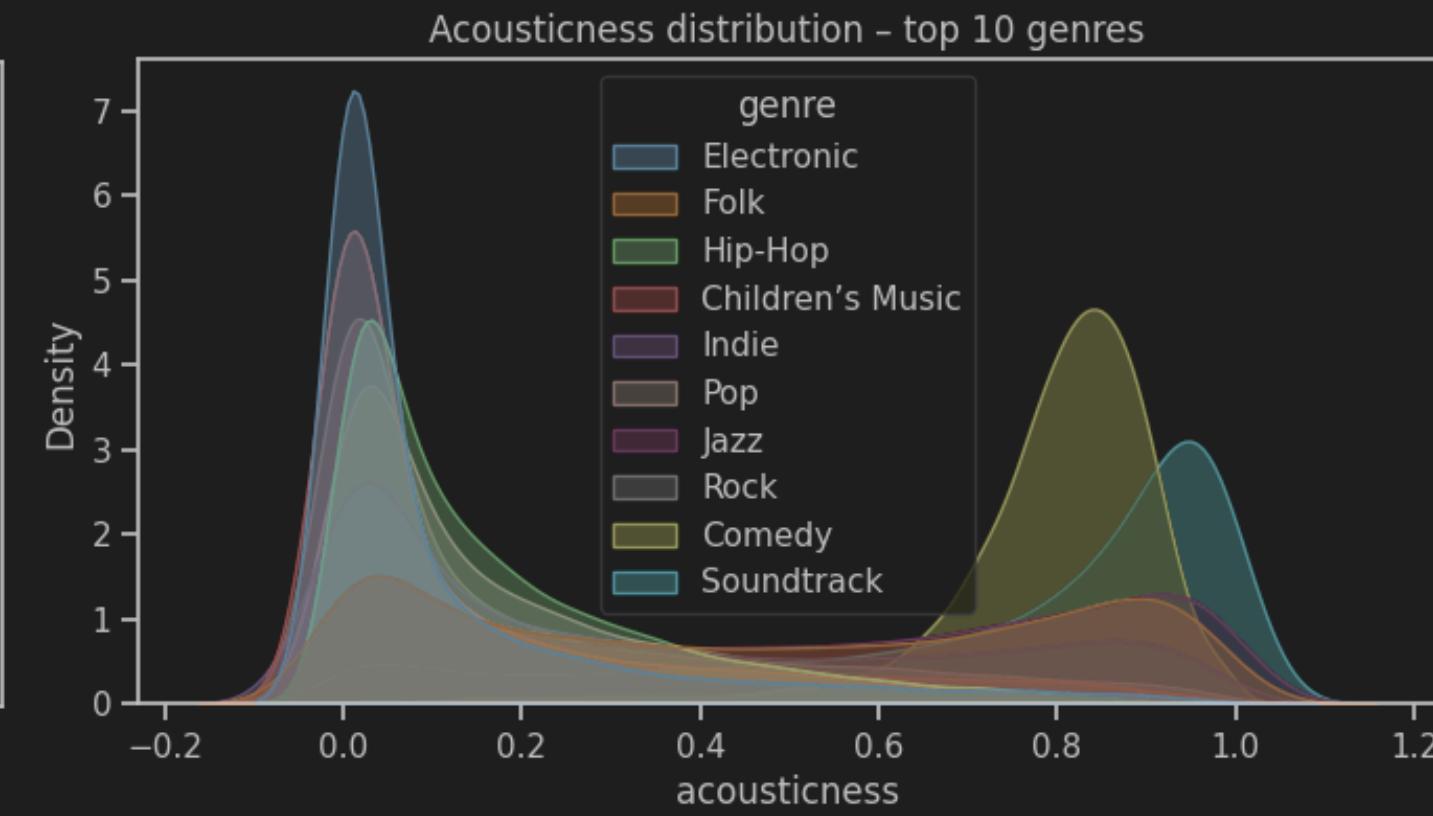
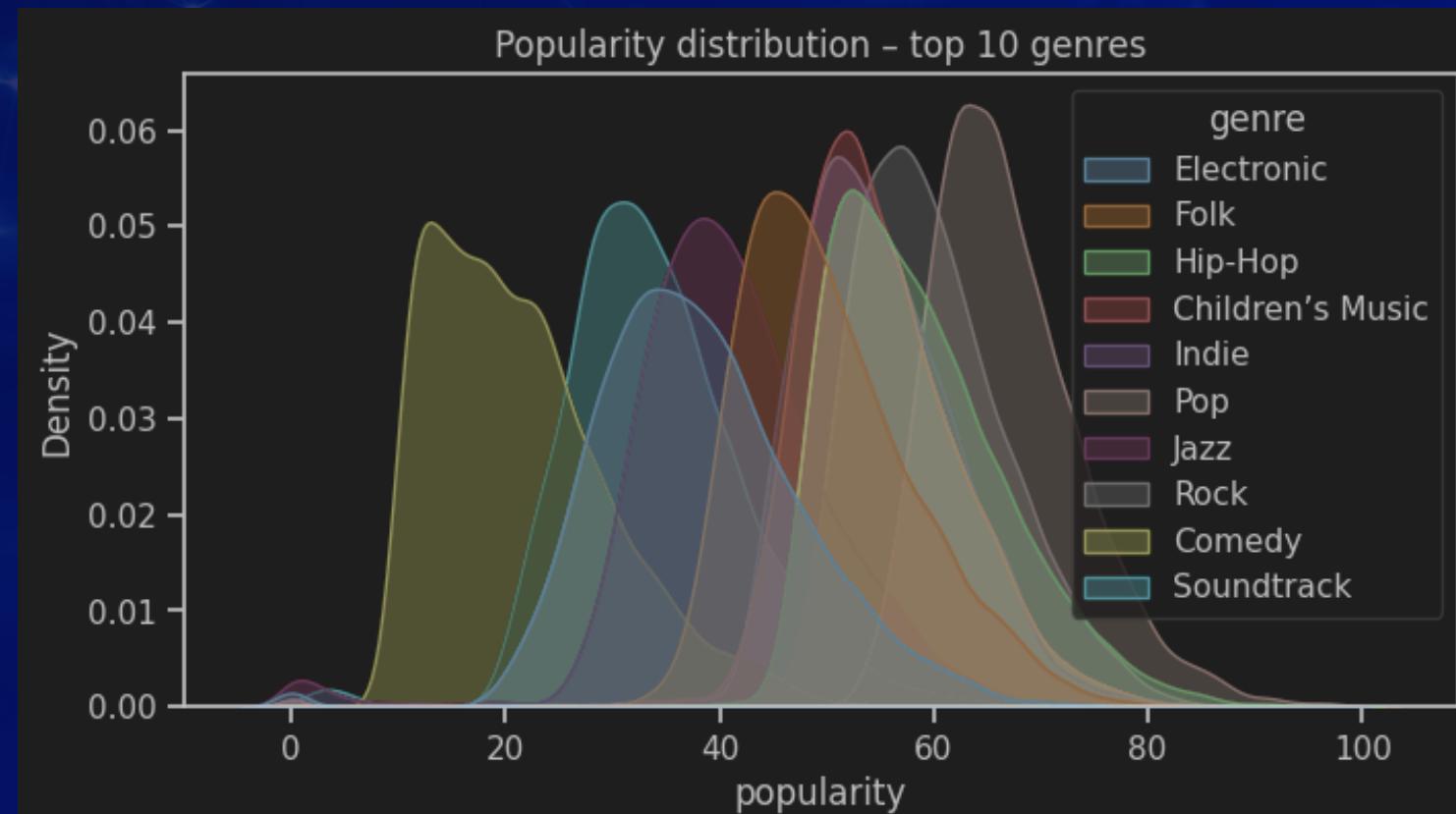
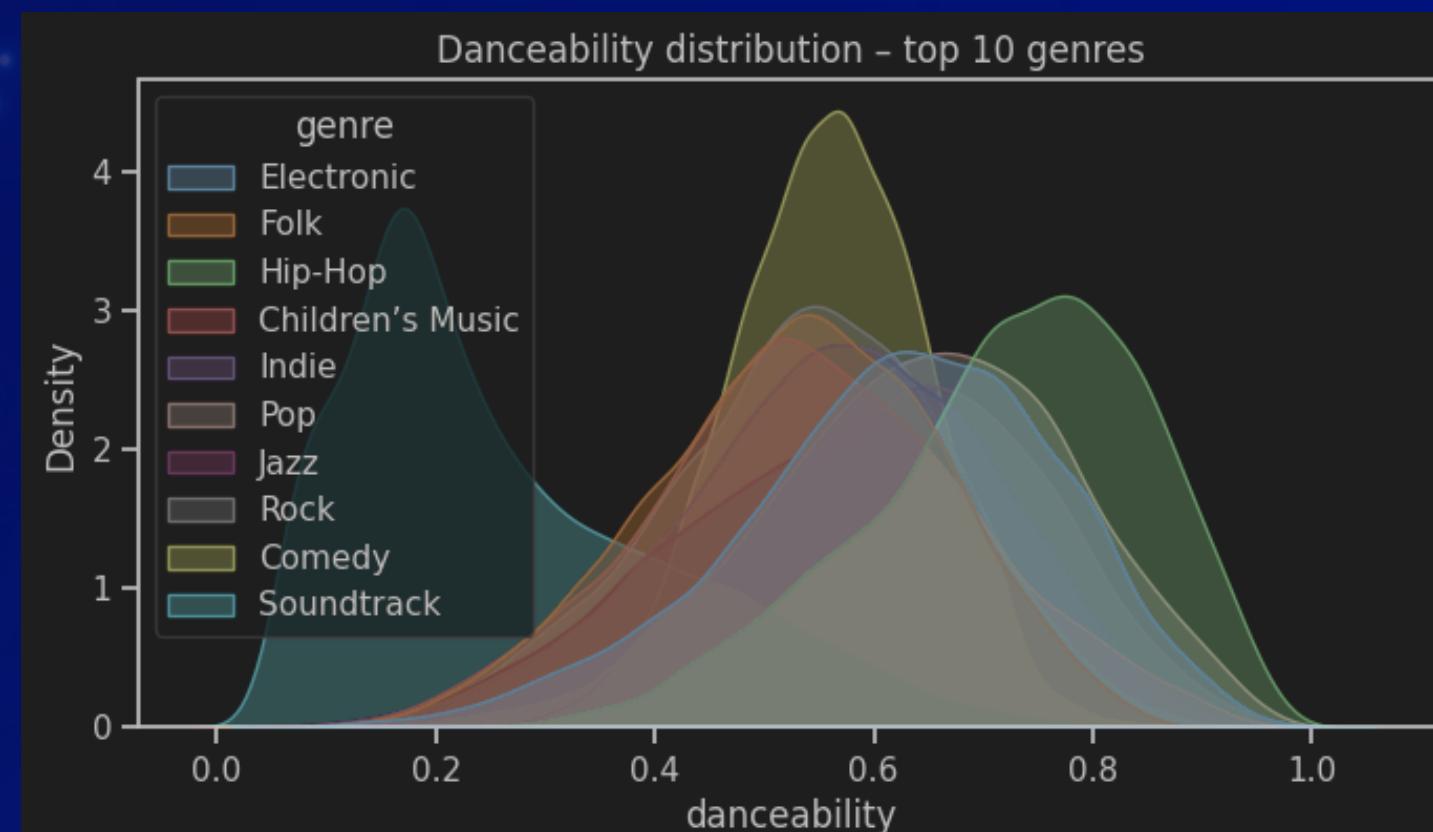
K < 1-11 > | 27 rows x 44 columns pd.DataFrame ↗

genre	popularity count	mean	median	std	acousticness count	mea
Comedy	9681	21.34	20.0	8.43	9681	
Soundtrack	9646	33.95	33.0	8.64	9646	
Indie	9543	54.70	54.0	7.36	9543	
Jazz	9441	40.82	40.0	9.59	9441	
Pop	9386	66.59	66.0	7.25	9386	
Electronic	9377	38.06	37.0	9.74	9377	
Children's Music	9353	54.66	54.0	7.85	9353	
Folk	9299	49.94	49.0	8.22	9299	
Hip-Hop	9295	58.42	57.0	8.27	9295	
Rock	9272	59.62	59.0	7.47	9272	
Alternative	9263	50.21	49.0	7.66	9263	

Distributions group by genre

Distribution of 10 genres feature values plotted
Only 10 because it makes a big mess to plot them all

hypothesis: those will be the most significant
features because it seems like every genre has its
range

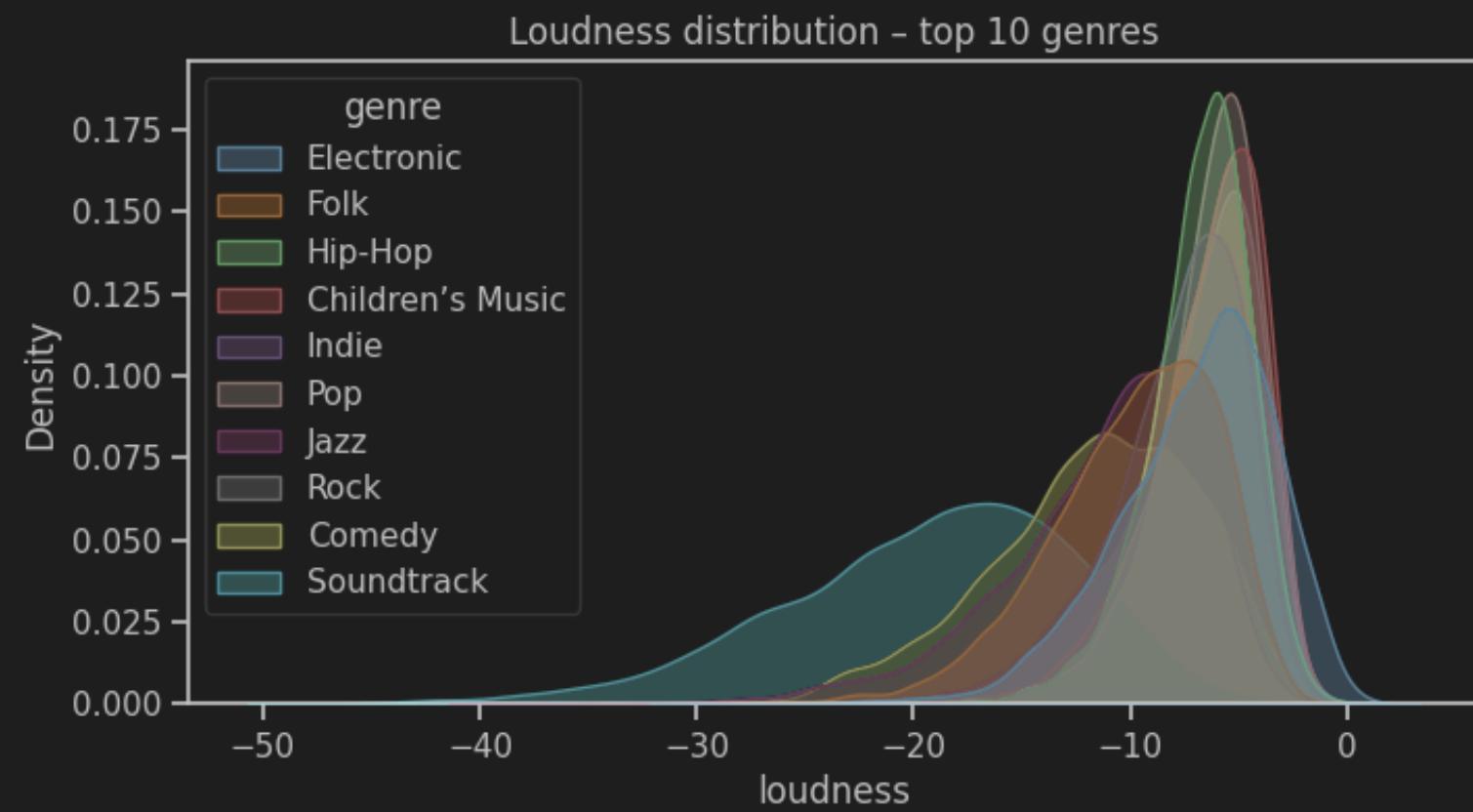
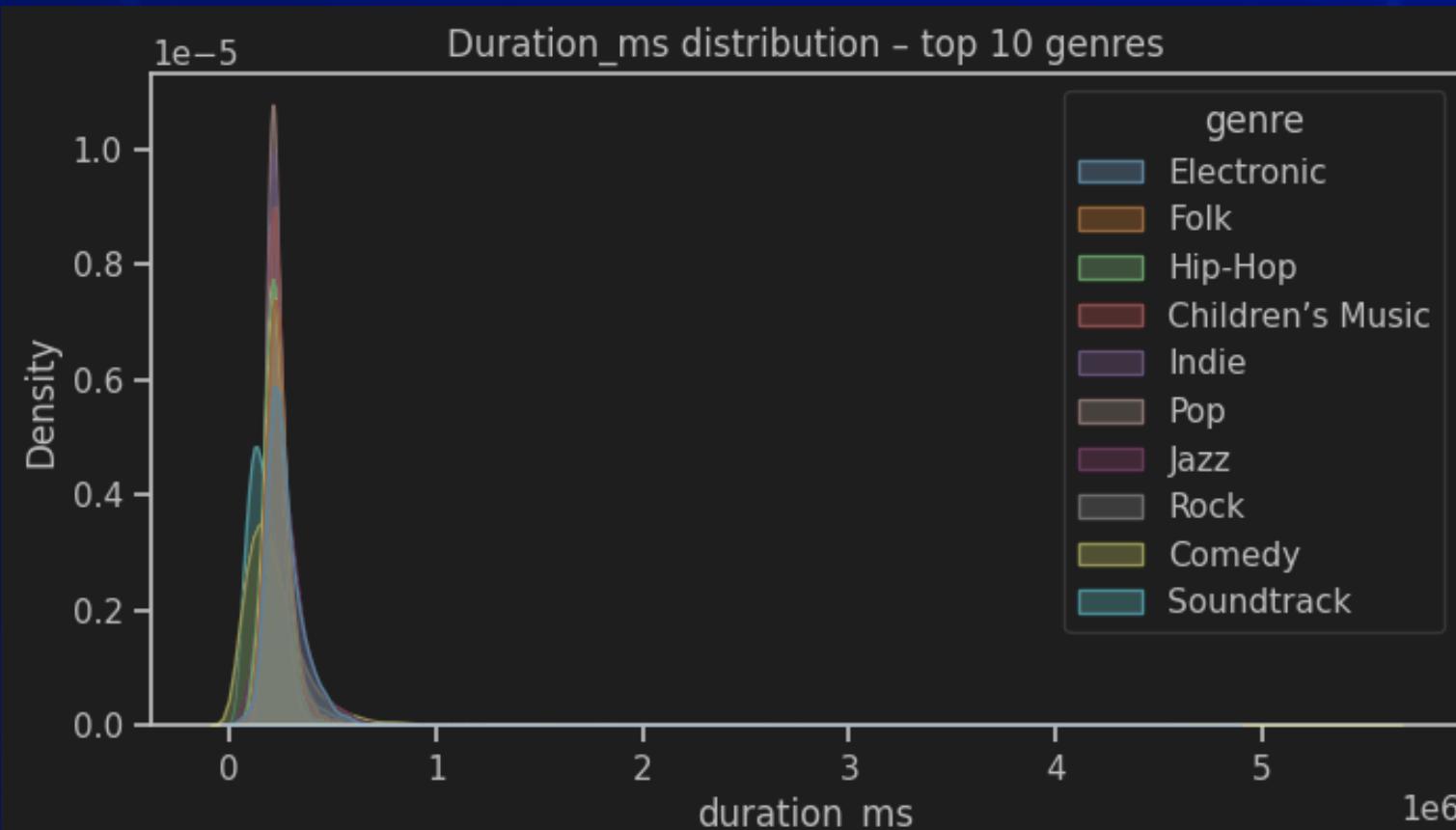
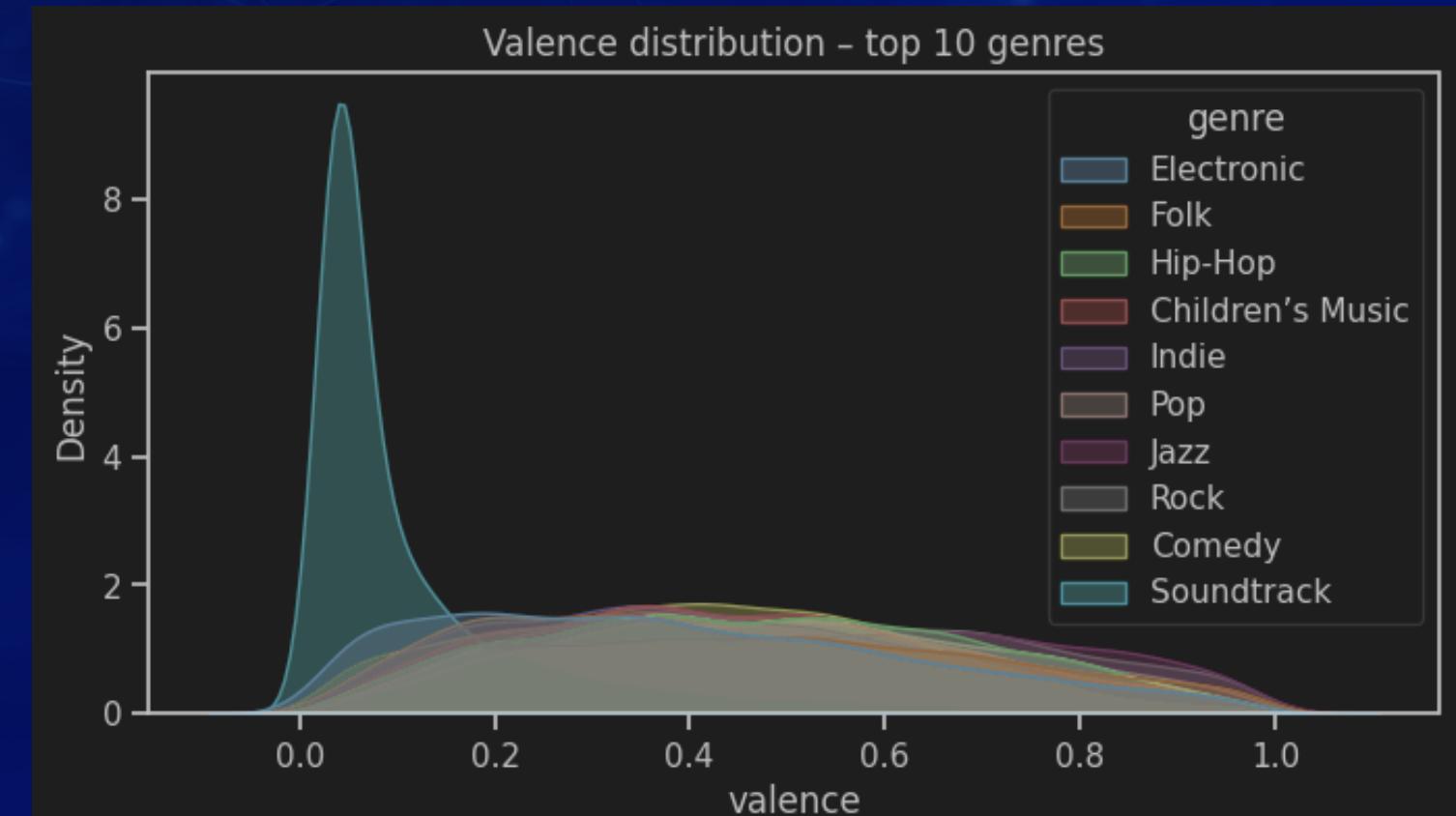


Lets talk genres!

Distribution of 10 genres feature values plotted
Only 10 because it makes a big mess to plot them all

Hypothesis: those will be the most insignificant
features because it seems all the genre ranges are
the same

I plotted all the distributions
it's all in the notebook



Just 4 Fun

Just because I was curious I wanted to see
The most popular artists in the dataset

It provides more information about
the data, but it's irrelevant

This information got me thinking about how
important the artist name to the genre classification

Top 15 consistently popular artists (≥ 25 tracks), JUST 4 FUN:

artist_name	mean_popularity	n_tracks
Post Malone	76.0	85
Juice WRLD	75.5	40
Offset	74.7	58
Khalid	74.4	32
XXXTENTACION	73.0	106
Metro Boomin	73.0	41
6ix9ine	72.9	26
Ed Sheeran	72.9	47
Ariana Grande	72.6	142
Hailee Steinfeld	72.3	25
Bad Bunny	72.0	34
Lil Peep	71.7	56
Alec Benjamin	71.7	25
Camila Cabello	71.0	50
Marshmello	70.8	34

Feature Engineering

Milliseconds to minutes:

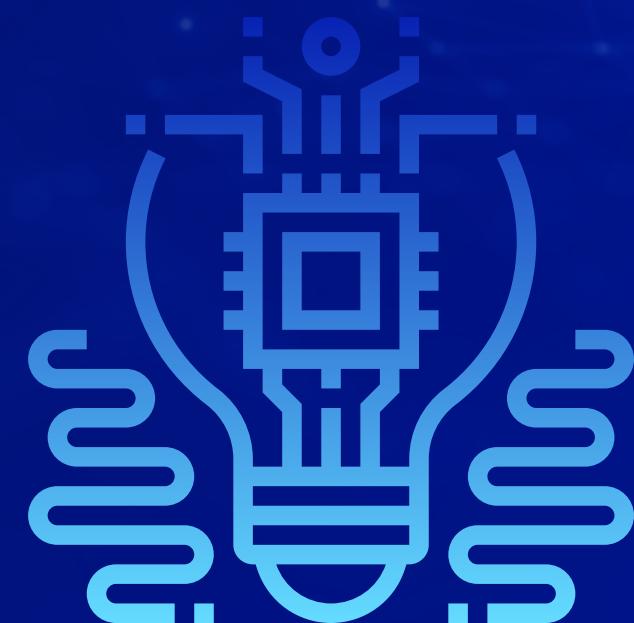
Milliseconds are not intuitive to the user.

The huge numbers will cause a longer training time.

Acoustic-instrumental synergy:

acousticness * instrumentalness

Might be useful for separating classic style from modern



Feature Engineering

Object features into numeric features:

Mode - binary 0 or 1

Key - mapping 1-12, then apply cos and sin encoding

Time signatures - numerator*2 / denominator

TF-IDF:

Artist name - apply mean TF-IDF

Track name - apply mean TF-IDF

Feature Engineering

Summary:

18 features in total all of them are 1D and numeric.

- "duration_min",
- "danceability", "energy", "liveness", "speechiness",
- "valence", "tempo", "acousticness",
- "instrumentalness", "loudness", "popularity",
- "ts_value", "is_major", "key_sin", "key_cos",
- "acoustic_instrumentalness",
- "track_tfidf_mean", "artist_tfidf_mean",

AdaBoost. Random-Forest. LightGBM.

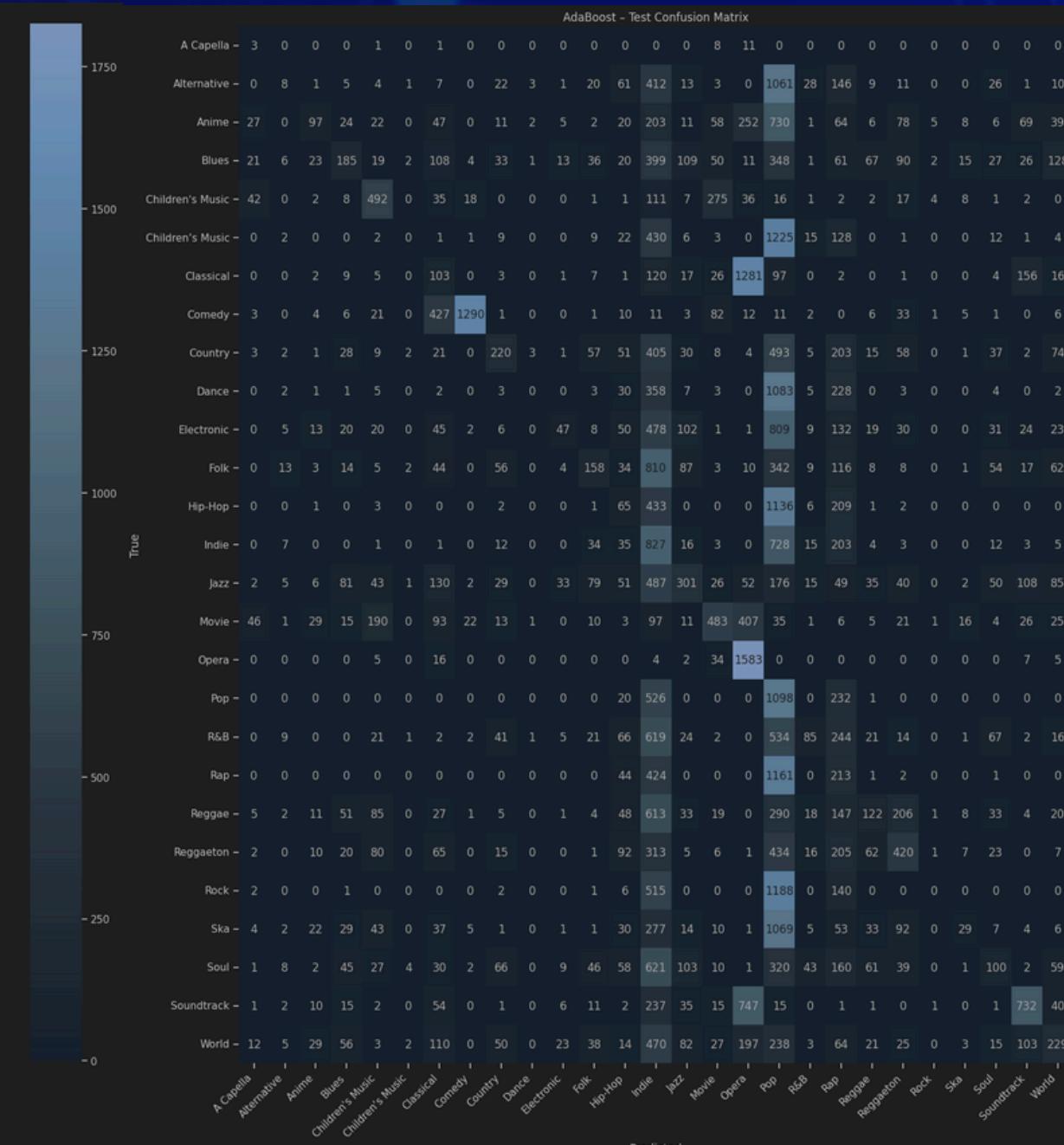
LightGBM
Random-Forest
AdaBoost

acc F1
42.5% / 42.7%
41.0% / 41.5%
20.4% / 19.0%

Random-Forest

Random-Forest - Test Confusion Matrix																																
True	A Capella	Alternative	Anime	Blues	Children's Music	Classical	Comedy	Country	Dance	Electronic	Folk	Hip-Hop	Indie	Jazz	Movie	Opera	Pop	R&B	Rap	Reggaeton	Rock	Ska	Soundtrack	World								
True	6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
A Capella	-	6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Alternative	-	0	170	6	44	0	756	0	0	71	63	52	58	65	89	15	1	0	28	130	33	16	14	136	21	49	2	34	0	0	0	
Anime	-	0	28	1154	41	3	6	35	1	20	2	45	2	0	0	9	62	37	1	2	0	24	22	9	99	4	126	55	0	0	0	
Blues	-	0	66	63	736	1	11	12	2	76	3	56	79	1	7	145	32	2	3	4	0	85	45	54	110	120	11	81	0	0	0	
Children's Music	-	0	0	3	8	920	0	2	3	1	0	1	1	0	0	1	111	2	0	1	0	3	9	0	12	3	0	0	0	0	0	0
Children's Music	-	0	492	0	9	1	54	1	1	46	110	17	61	103	180	10	0	0	53	165	78	6	5	353	7	57	0	62	0	0	0	
Classical	-	0	22	36	22	7	1	1269	0	3	0	24	7	0	1	56	19	110	0	0	0	1	0	2	6	4	215	46	0	0	0	
Comedy	-	0	0	2	11	17	2	0	1826	1	0	0	4	1	0	1	41	1	0	0	0	7	5	1	4	7	0	5	0	0	0	0
Country	-	0	42	12	66	3	28	2	1	832	30	9	175	7	19	21	13	0	57	20	26	25	47	198	16	37	1	46	0	0	0	
Dance	-	0	52	2	1	5	106	0	0	76	180	59	42	107	82	9	11	0	421	263	107	11	30	120	7	44	0	5	0	0	0	
Electronic	-	0	72	37	55	1	18	3	1	11	50	1050	22	7	10	205	4	0	12	16	4	66	53	13	25	38	34	68	0	0	0	
Folk	-	0	59	4	44	0	32	2	1	215	45	22	407	5	375	99	6	1	77	36	0	15	7	256	22	74	21	35	0	0	0	
Hip-Hop	-	0	17	0	0	0	78	0	1	4	85	11	3	303	23	9	0	0	183	163	839	13	62	17	0	48	0	0	0	0	0	
Indie	-	0	86	0	4	0	179	0	0	57	133	13	474	50	98	18	0	1	137	177	58	12	7	273	4	114	3	11	0	0	0	
Jazz	-	0	22	7	176	7	5	44	0	33	6	241	107	15	19	721	26	0	7	27	0	50	39	9	9	200	56	62	0	0	0	
Movie	-	0	0	28	32	67	0	29	13	16	1	1	15	2	1	10	1084	108	3	6	1	5	11	1	20	4	78	25	0	0	0	
Opera	-	0	0	12	6	0	0	93	0	0	0	0	2	0	0	1	53	1471	0	0	0	0	0	0	0	0	0	12	6	0	0	0
Pop	-	0	4	0	1	0	43	0	0	39	388	9	51	214	116	1	1	0	218	131	358	0	21	255	1	20	4	2	0	0	0	
R&B	-	0	57	2	3	0	127	0	0	43	272	12	41	146	140	34	4	0	153	167	119	32	48	9	1	374	1	13	0	0	0	
Rap	-	0	15	0	0	0	83	0	0	25	123	4	2	918	41	8	0	0	309	124	97	0	14	67	0	14	1	1	0	0	0	
Reggae	-	0	25	42	73	10	13	0	2	30	25	57	14	48	12	40	15	0	5	46	6	783	218	44	179	37	3	27	0	0	0	
Reggaeton	-	0	8	33	38	2	1	0	1	29	32	40	3	111	3	8	9	0	62	45	25	151	1120	4	27	22	0	11	0	0	0	
Rock	-	0	65	3	38	0	336	1	0	170	100	3	246	19	242	3	1	0	229	10	46	26	3	242	7	32	8	25	0	0	0	
Ska	-	0	33	111	99	8	16	2	4	29	12	32	3	1	5	15	19	0	1	1	0	216	49	16	1080	9	2	12	0	0	0	
Soul	-	0	34	3	136	2	28	2	2	89	64	79	112	52	135	213	6	0	34	394	20	83	51	47	4	195	5	28	0	0	0	
Soundtrack	-	0	0	64	18	0	0	150	0	0	0	20	7	0	6	35	72	7	1	1	0	0	2	1	1507	37	0	0	0	0	0	
World	-	0	34	53	115	0	36	65	4	66	0	130	56	0	3	130	17	8	2	2	0	30	26	14	19	23	140	846	0	0	0	

***pop + indie



PCA:

My hypothesis:
Strong features-

popularity

danceability

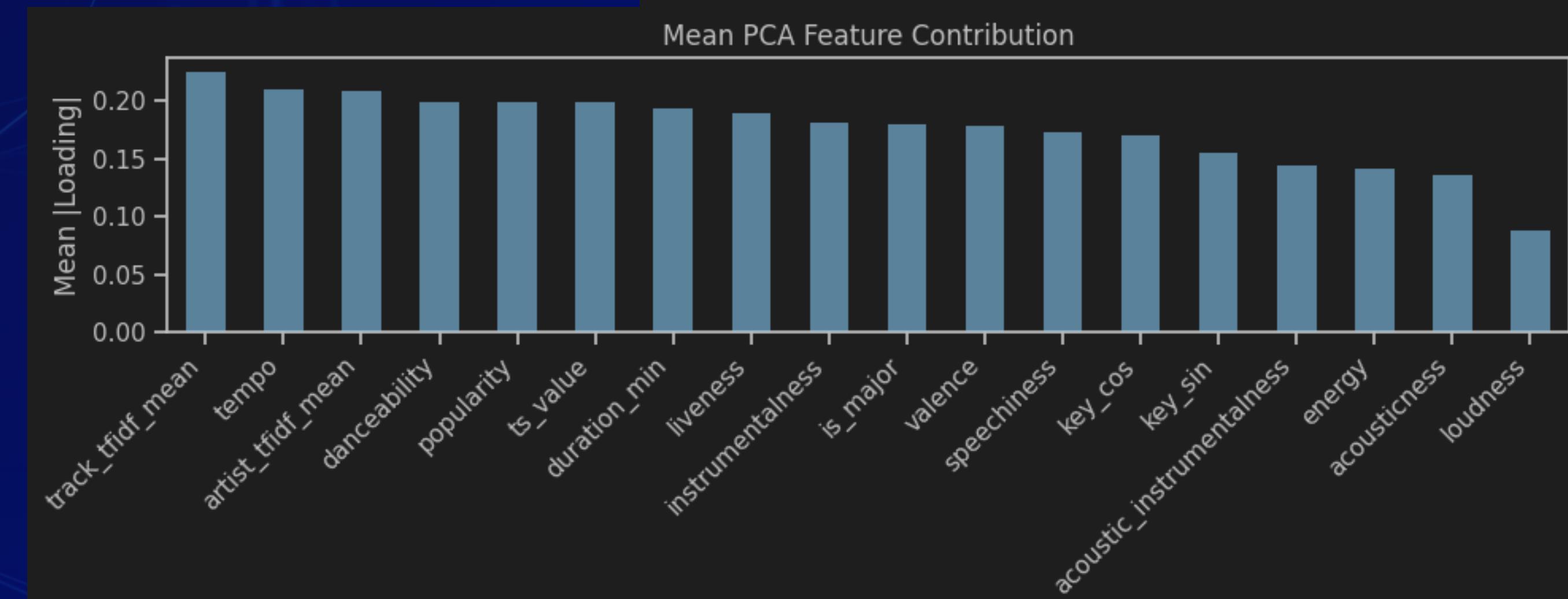
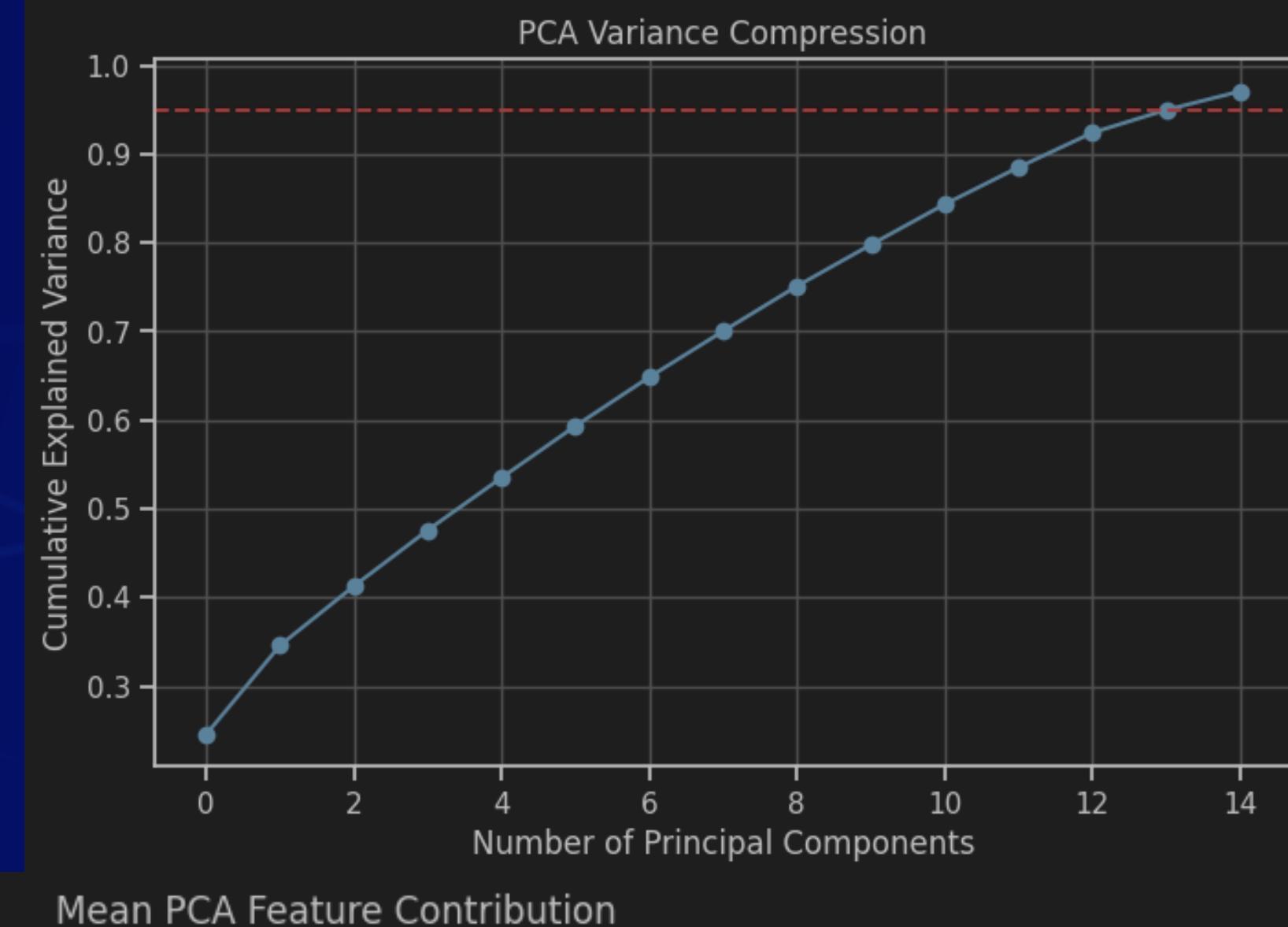
acousticness

Weak features-

valence

loudness

Original numeric features: 18
PCA kept components: 15



PCA: AdaBoost. Random-Forest.

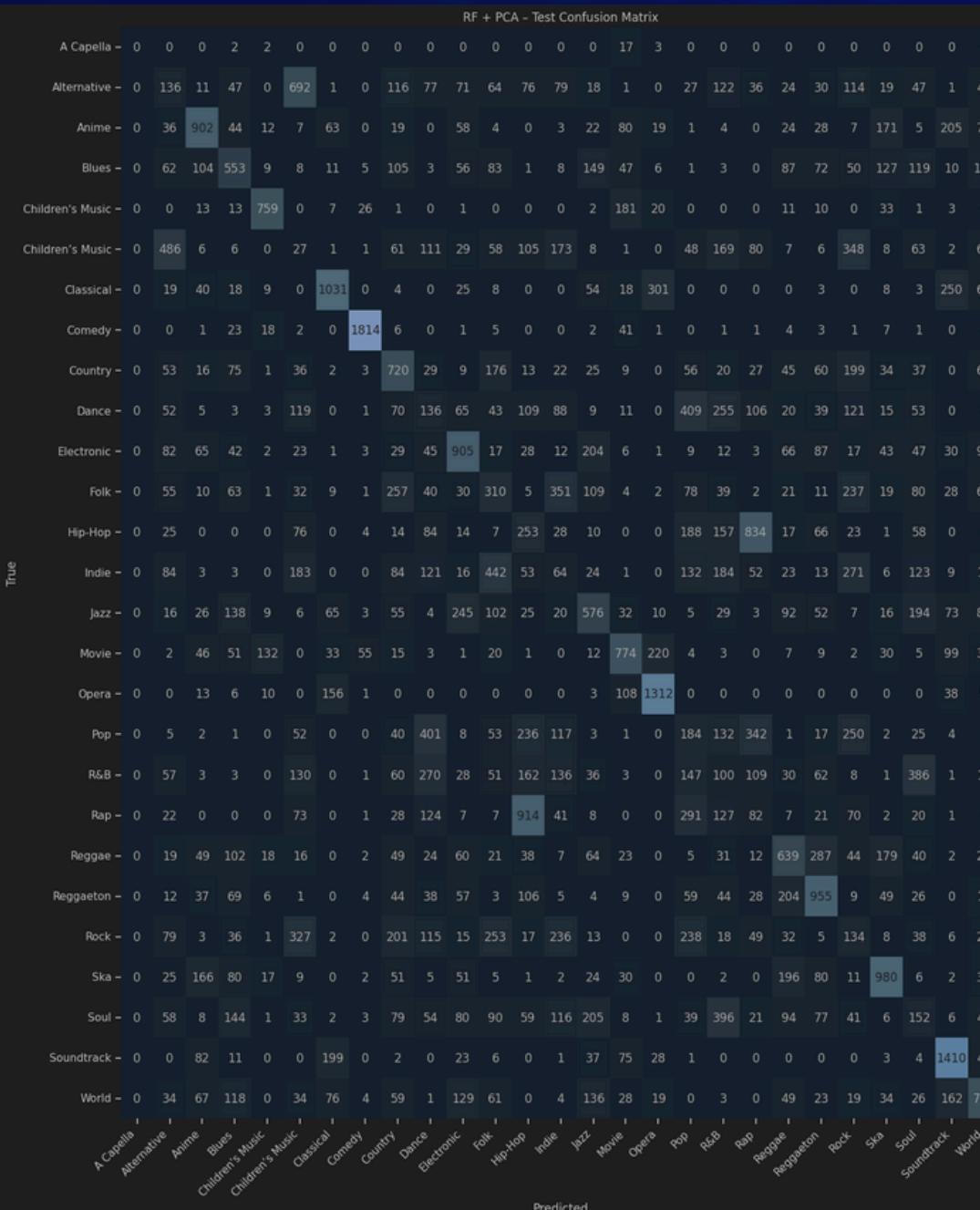
Random Forest + PCA
AdaBoost + PCA

acc F1
34.0% / 33.7%
17.3% / 14.5%

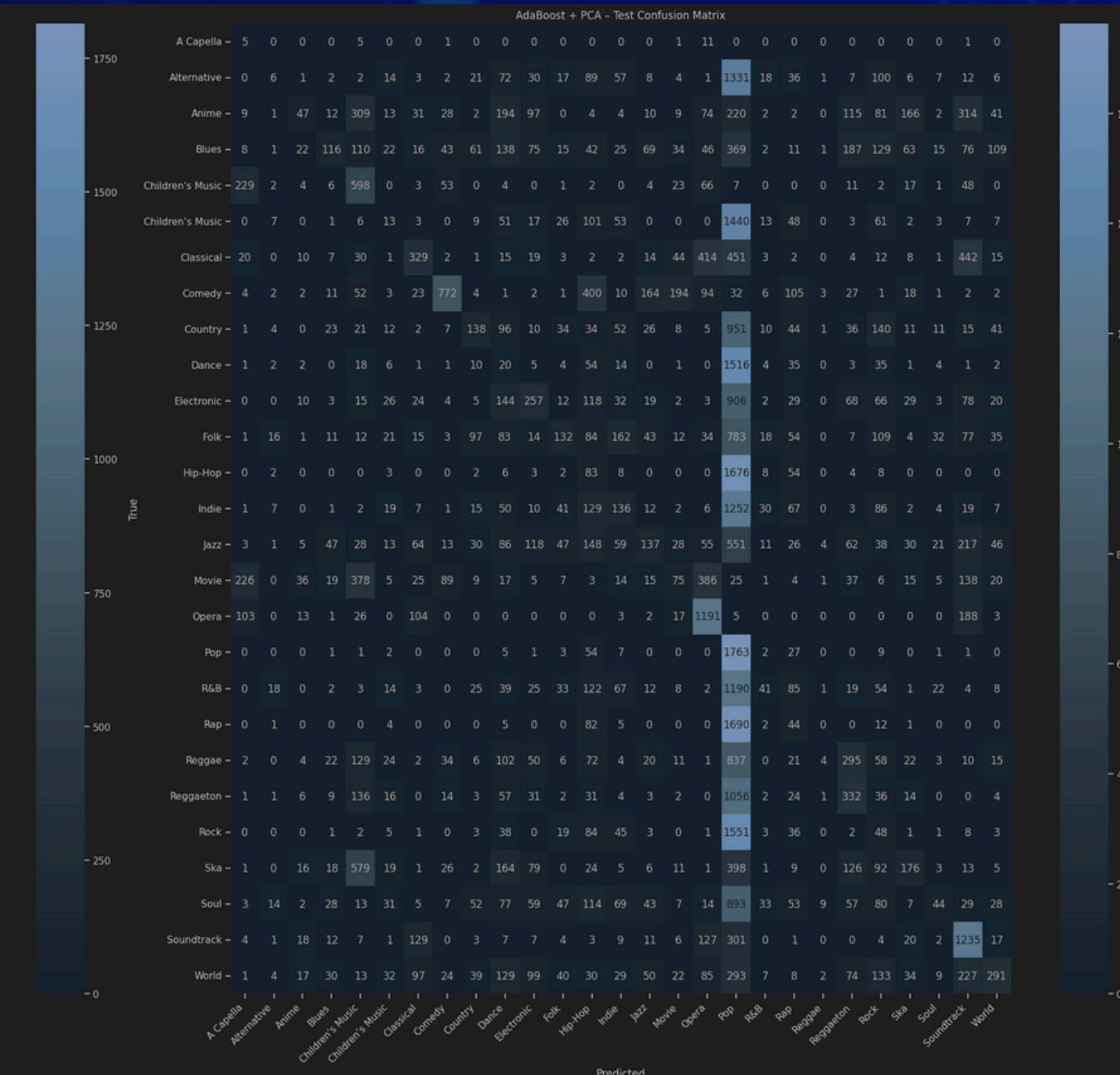
I wasn't surprised that
the performance is lower.

But I didn't expect such a drop

Random-Forest



AdaBoost



K-mean: AdaBoost. Random-Forest.

Run the data through K-means
algorithm with $K = 8$

I chose 8 and not 27 because want the
preprocessing to help the models not
to be a model by itself.

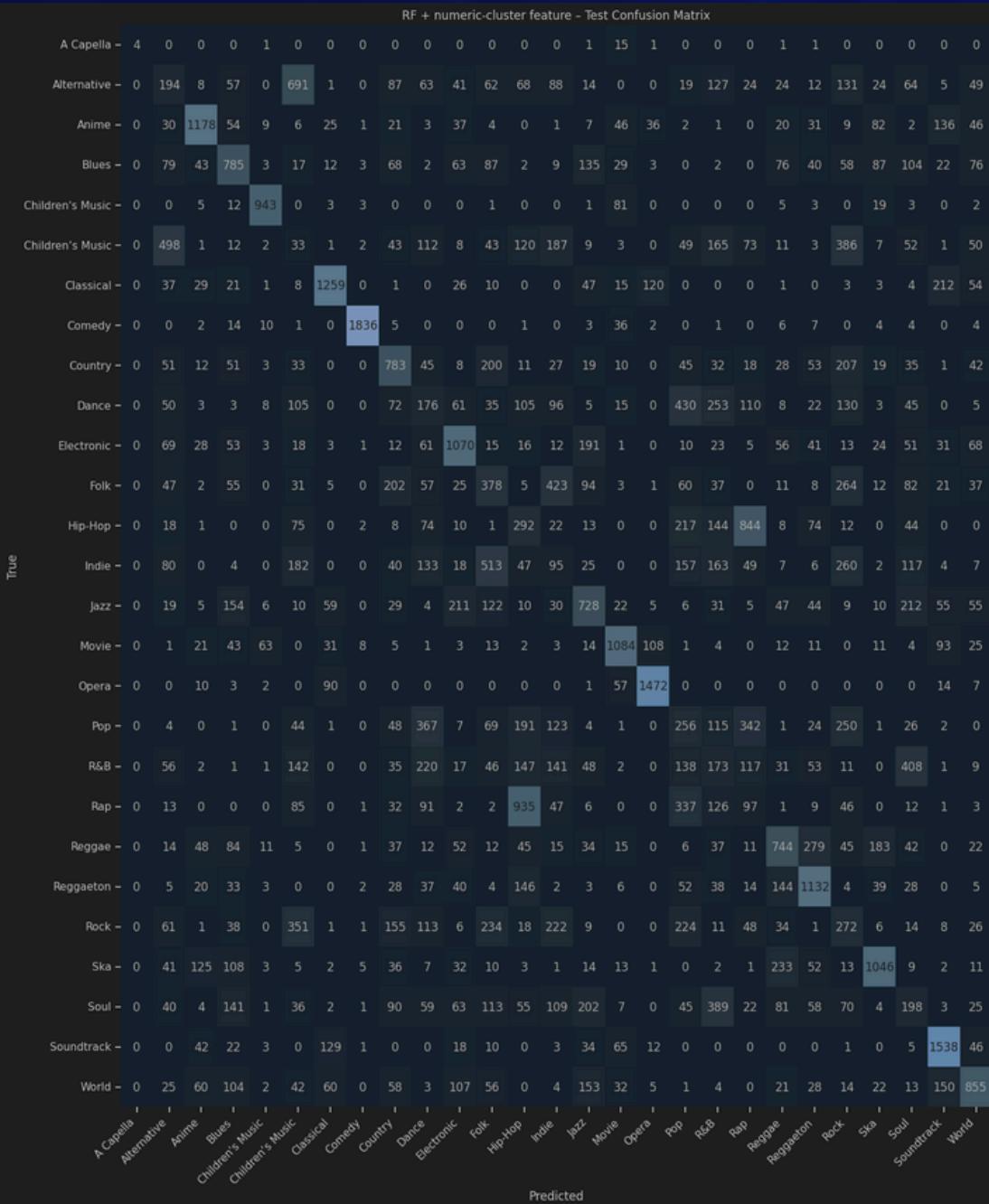
K-mean: AdaBoost. Random-Forest.

RandomForest KMeans
AdaBoost + KMeans

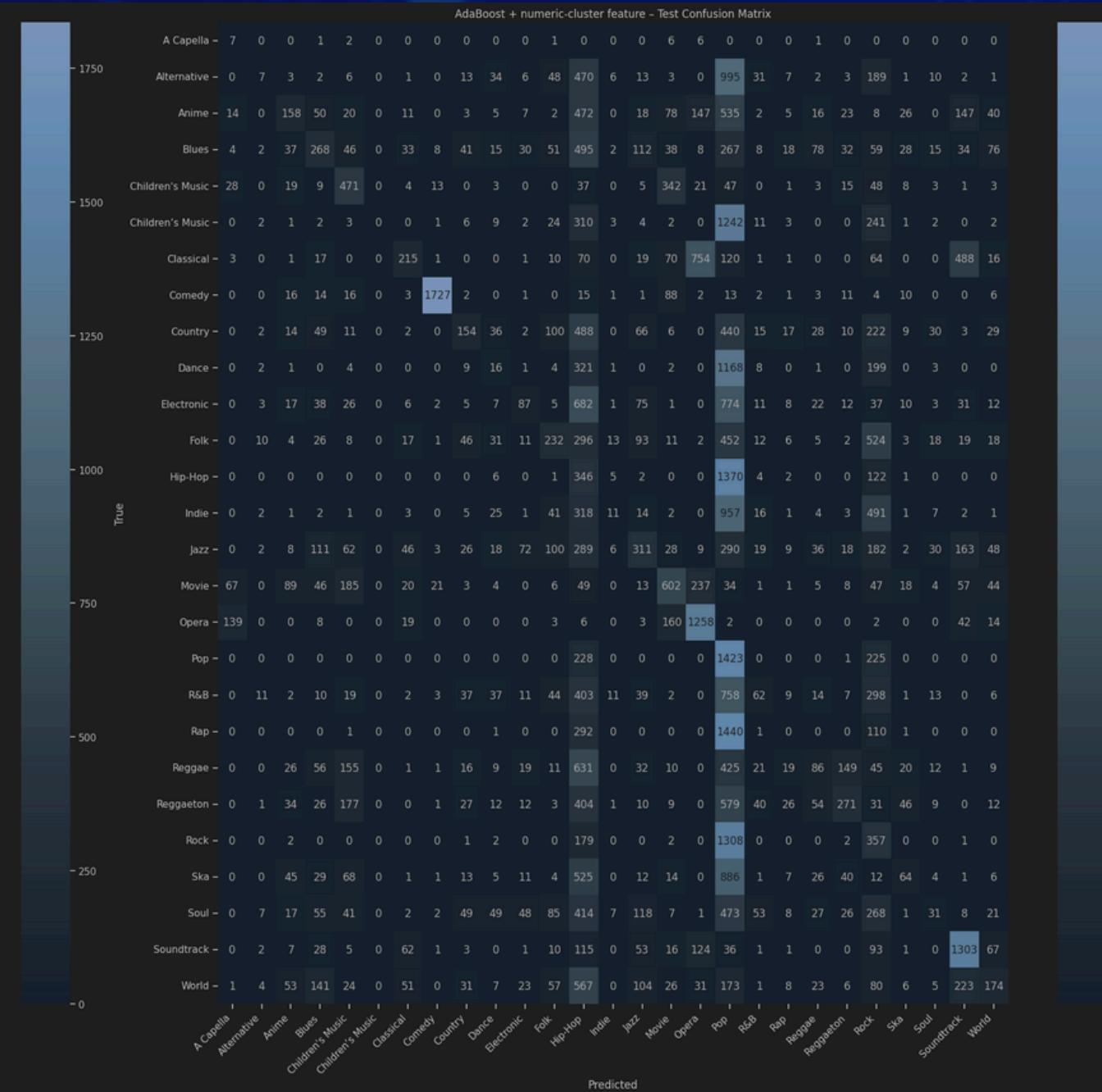
acc F1
41.1% / 41.9%
20.2% / 19.2%

The extra feature improves the performance slightly.

Random-Forest



AdaBoost



Summery

LightGBM	42.5% / 42.7%
Random-Forest + KMeans feature	41.1% / 41.9%
Random-Forest baseline	41.0% / 41.5%
Random-Forest + PCA	34.0% / 33.7%
AdaBoost	20.4% / 19.0%
AdaBoost + KMeans feature	20.2% / 19.2%
AdaBoost + PCA	17.3% / 14.5%

Thanks!

It was a pleasure to work on this project.