In [1]:
```
'''

Method: Run analysis on dataset.

Parameters: Playlist_dataset, Track_dataset

Body:

Shows charts and visualisations for the data in both datasets.
Histograms, distributions
Basic metrics

'''
```

Out[1]:
```
'\n\nMethod: Run analysis on dataset.\n\nParameters: Playlist_dataset
, Track_dataset\n\nBody:\n\nShows charts and visualisations for the d
ata in both datasets.\nHistograms, distributions\nBasic metrics\n\n'
```

In [403]:
```
import json
import pandas as pd
import plotly.express as px
import plotly.graph_objects as go
import numpy as np
from collections import defaultdict
import copy
from tqdm import tqdm
from sklearn.metrics import ndcg_score
# import re
# import plotly.io as pio
# pio.renderers.default='notebook'
```

In [ ]:
```
!pip3 install pandas
```

In [ ]:
```
def parseData(fname):
    for l in open(fname):
        yield eval(l)
```

In [ ]:
```
with open('mpd.slice.0-999.json') as f:
    data = json.load(f)
```

In [ ]:
```
data["info"]
```

In [ ]:
```
data["playlists"]
```

In [ ]:
```
playlists = data['playlists']
playlists_df = pd.DataFrame(playlists)
```

In [ ]:
```
# tracks = {}
# albums = {}
# artists = {}
# for t in playlists_df['tracks']:
#     for t_ in t:
#         tracks[t_['track_uri']] = t_
#         artists[t_['artist_uri']] = t_
#         albums[t_['album_uri']] = t_

# tracks_df = pd.DataFrame(tracks.values(), index=tracks.keys())
```

In [ ]:
```
playlists_df.head(1)['name']
```

In [28]:
```
def loadPlaylistData(data):
    under_60_tracks_playlists = [playlist for playlist in data['playli
    tracks = []
    for playlist in under_60_tracks_playlists:
        track_uris = []
        for track in playlist['tracks']:
            tracks.append(track)
            track_uris.append(track['track_uri'])
        playlist['tracks'] = track_uris
    p_df = pd.DataFrame(under_60_tracks_playlists).set_index('pid', dr
    t_df = pd.DataFrame(tracks).set_index('track_uri', drop=False)
    return under_60_tracks_playlists, p_df, t_df
```

In [29]:
```
l_playlists = pd.DataFrame()
l_tracks = pd.DataFrame()
= []

r fileName in ['mpd.slice.122000-122999.json', 'mpd.slice.308000-30899
  f = open(fileName)
  data = json.load(f)
  l, playlist_df, tracks_df = loadPlaylistData(data)
  # print(playlist_df)
  L.append(l)
  all_playlists = pd.concat([all_playlists, playlist_df])
  all_tracks = pd.concat([all_tracks, tracks_df])

playlists_df1 = pd.concat(all_playlists)
tracks_df1 = pd.concat(all_tracks)
l_tracks = all_tracks.drop_duplicates(subset=['track_uri'])
all_tracks.set_index('track_uri')
```

In [ ]:
```
# tracks_csv = pd.read_csv('track_data.csv', sep='\t')
# playlists_csv = pd.read_csv('playlist_data.csv', sep='\t')
```

In [ ]:
```
# all_playlists = playlists_csv
# all_tracks = tracks_csv
```

In [144]:
```
a, mp_playlists, mp_tracks, d, e = showDataAnalysis_new(all_playlists,
```

In [ ]:
```
# showDataAnalysis(playlists_df, tracks_df)
```

In [90]:
```
def showDataAnalysis_new(playlists_df, tracks_df):
#     tracks = {}
    albums = {}
    artists = {}
    track_popularity = defaultdict(int)
    album_popularity = defaultdict(int)
    artist_popularity = defaultdict(int)


    for t in playlists_df['tracks']:
#         ts = t.split('\'')[1::2]
        for t_ in t:
#             tracks[t_['track_uri']] = t_
            track = tracks_df[tracks_df['track_uri'] == t_]
#             print (t_, track['artist_uri'])
#             print (track.iloc[0], type(track))
#             print ("S", track['artist_uri'])
            artists[track['artist_uri'].iloc[0]] = track.iloc[0]
            albums[track['album_uri'].iloc[0]] = track.iloc[0]

            track_popularity[t_] += 1
            album_popularity[track['album_uri'].iloc[0]] += 1
            artist_popularity[track['artist_uri'].iloc[0]] += 1

#     tracks_df = pd.DataFrame(tracks.values(), index=tracks.keys())
    artists_df = pd.DataFrame(artists.values(), index=artists.keys())
    albums_df = pd.DataFrame(albums.values(), index=albums.keys())

#     track_popularity = defaultdict(int)
#     album_popularity = defaultdict(int)
#     artist_popularity = defaultdict(int)

#     for i in playlists_df['tracks']:
#         for track in i:
#             track_popularity[track] += 1
#             album_popularity[tracks_df[tracks_df['track_uri'] == tra
#             artist_popularity[tracks_df[tracks_df['track_uri'] == tr
```

```
    tracks_df['popularity'] = tracks_df['track_uri'].map(track_popular
    artists_df['popularity'] = artists_df['artist_uri'].map(artist_pop
    albums_df['popularity'] = albums_df['album_uri'].map(album_popular

    mostPopularPlaylist = playlists_df.sort_values(by='num_followers',
    mostPopularTracks = tracks_df.sort_values(by=['popularity'], ascen
    mostPopularAlbums = albums_df.sort_values(by=['popularity'], ascen
    mostPopularArtists = artists_df.sort_values(by=['popularity'], asc

#     printCharts(playlists_df, mostPopularPlaylist, mostPopularTracks
    return playlists_df, mostPopularPlaylist, mostPopularTracks, mostP
```

In [ ]:
```
printCharts(a, b, c, d, e)
```

In [ ]:
```
def printCharts(playlists_df, mostPopularPlaylist, mostPopularTracks,

    print ("Charts/Graphs: ")

    # Scatter - num_tracks
    fig = go.Figure()
    fig.add_trace(go.Scatter(x=playlists_df.index, y=playlists_df['num
                             mode='markers',
                             name='markers'))
    fig.add_trace(go.Scatter(x=playlists_df.index, y=[playlists_df['nu
                             mode='lines',
                             name='lines'))

    fig.update_layout(
        title="Scatter plot for number of tracks in each playlist and
        xaxis_title="Playlist index",
        yaxis_title="num_tracks")

    fig.show()

    # Scatter - num_albums
    fig = go.Figure()
    fig.add_trace(go.Scatter(x=playlists_df.index, y=playlists_df['num
                             mode='markers',
                             name='markers'))
    fig.add_trace(go.Scatter(x=playlists_df.index, y=[playlists_df['nu
                             mode='lines',
                             name='lines'))

    fig.update_layout(
        title="Scatter plot for number of albums in each playlist and
        xaxis_title="Playlist index",
        yaxis_title="num_albums")
    fig.show()
```

```
# Histogram - num_edits
fig = px.histogram(playlists_df, x="num_edits")
fig.update_layout(
    title="Histogram of number of albums in the playlists",
    xaxis_title="num_albums bins",
    yaxis_title="occurence_count")
fig.show()

# Histogram - num_tracks
fig = px.histogram(playlists_df, x="num_tracks")
fig.update_layout(
    title="Histogram of number of tracks in the playlists",
    xaxis_title="num_tracks bins",
    yaxis_title="occurence_count")
fig.show()

# Histogram - num_followers
fig = px.histogram(playlists_df, x="num_followers")
fig.update_layout(
    title="Histogram of number of followers for the playlists",
    xaxis_title="num_followers bins",
    yaxis_title="occurence_count")
fig.show()

# Histogram - duration_ms
fig = px.histogram(playlists_df, x="duration_ms")
fig.update_layout(
    title="Histogram of duration (in ms) of the playlists",
    xaxis_title="duration (in ms) bins",
    yaxis_title="occurence_count")
fig.show()

# Scatter - Playlists with relation b/w duration_ms and num_follow
fig = px.scatter(playlists_df, x=playlists_df.index, y='duration_m
fig.update_layout(
    title="Correlation of number of followers with duration (in ms
    xaxis_title="playlist index",
    yaxis_title="duration_ms")
fig.show()

# Scatter - Most popular playlists
fig = px.bar(mostPopularPlaylist[:20], x='name', y='num_followers'
fig.update_layout(
    title="Most popular playlists",
    xaxis_title="playlist index",
    yaxis_title="num_followers")
fig.show()

# Scatter - Most Popular Tracks
fig = px.bar(mostPopularTracks[:20], x='track_name', y='popularity
fig.update_layout(
```

```
    fig.update_layout(
        title="Most popular tracks",
        xaxis_title="playlist index",
        yaxis_title="track popularity")
    fig.show()

    # Scatter - Most popular Albums
    fig = px.bar(mostPopularAlbums[:20], x='album_name', y='popularity
    fig.update_layout(
        title="Most popular albums",
        xaxis_title="playlist index",
        yaxis_title="album popularity")
    fig.show()

    # Scatter - Most popular Artists
    fig = px.bar(mostPopularArtists[:20], x='artist_name', y='populari
    fig.update_layout(
        title="Most popular artists",
        xaxis_title="playlist index",
        yaxis_title="artist popularity")
    fig.show()
```

In [ ]:

In [14]:

```
pos
artist_name
track_uri
artist_uri
track_name
album_uri
duration_ms
album_name
```

Heuristic Models

In [30]:
```
all_playlists.head(10)
```

Out[30]:

pid	name	collaborative	pid	modified_at	num_tracks	num_albums	num_followers
122000	drum & bass	false	122000	1460332800	12	11	1
122001	chillllll	false	122001	1466899200	29	28	1
122004	November	false	122004	1416787200	40	30	1
122007	spring 2016	false	122007	1465430400	53	39	1
122008	Emily	false	122008	1422230400	16	16	1
122010	feel good	false	122010	1505692800	23	22	1
122012	oldie but a goodie	false	122012	1499126400	31	20	2
122013	Spring	false	122013	1377216000	33	30	1
122014	Yaa	false	122014	1493942400	40	36	2
122015	Vibes	false	122015	1508544000	47	35	2

In [592]:
```
trainData = []
testData = []
c = 0
playlistsPerTrack = defaultdict(set)
tracksPerPlaylist = defaultdict(set)
tracksPerArtist = defaultdict(set)
artistsPerTrack = defaultdict(set)

for idx, playlist in all_playlists.iterrows():
    c += 1
#     if c == 11:
#         break
    tempPlaylist = copy.deepcopy(playlist)
    tempTracks = []

#     print (idx)
    for i in range(len(playlist['tracks'])):
        tr = all_tracks.loc[playlist['tracks'][i]]
        tempTracks.append(playlist['tracks'][i])
        playlistsPerTrack[playlist['tracks'][i]].add(idx)
        tracksPerPlaylist[idx].add(playlist['tracks'][i])
        artistsPerTrack[playlist['tracks'][i]].add(tr['artist_uri'])
        tracksPerArtist[tr['artist_uri']].add(playlist['tracks'][i])

    tempPlaylist['tracks'] = tempTracks
    trainData.append(tempPlaylist)

#     p = copy.deepcopy(tempPlaylist)
#     t = [playlist['tracks'][-1]]
#     p['tracks'] = t
#     testData.append(p)
```

In [89]:
```
y_train = [t['tracks'][0] for t in testData]
len(y_train)
```

Out[89]: 2656

In [44]:
```
trainData[0]['tracks']
```

Out[44]:
```
['spotify:track:35AmCchFjTJoVwymcRmiLC',
 'spotify:track:2QePQ29ix8gC0CbRHcGoBz',
 'spotify:track:7zbq8RT5Kd3ExOGVTiUQbR',
 'spotify:track:3d5LFzH1i152LuEUxNZJAD',
 'spotify:track:1El3MOnnIhaT6MsRqNu13q',
 'spotify:track:5QHC8Li12dGunS3bq9Pu25',
 'spotify:track:0ay5L8O5lSsK0KvzJ73ysp',
 'spotify:track:30s0O3paoTyMfHoUNBQfa0',
 'spotify:track:0EnhRBCBQ0ldmkbTnXCTUU',
 'spotify:track:5ne28OLLS7xUvVK8nNe6Yg',
 'spotify:track:73FNl823P9rJQ4r3yNyOI8']
```

In [52]:
```
tracksPerPlaylist[122000]
```

Out[52]:
```
{'spotify:track:0EnhRBCBQ0ldmkbTnXCTUU',
 'spotify:track:0ay5L8O5lSsK0KvzJ73ysp',
 'spotify:track:1El3MOnnIhaT6MsRqNu13q',
 'spotify:track:2QePQ29ix8gC0CbRHcGoBz',
 'spotify:track:30s0O3paoTyMfHoUNBQfa0',
 'spotify:track:35AmCchFjTJoVwymcRmiLC',
 'spotify:track:3d5LFzH1i152LuEUxNZJAD',
 'spotify:track:5QHC8Li12dGunS3bq9Pu25',
 'spotify:track:5ne28OLLS7xUvVK8nNe6Yg',
 'spotify:track:73FNl823P9rJQ4r3yNyOI8',
 'spotify:track:7zbq8RT5Kd3ExOGVTiUQbR'}
```

In [63]:
```
all_playlists.loc[122001]['tracks']
```

Out[63]:
```
['spotify:track:1wZqJM5FGDEl3FjHDxDyQd',
 'spotify:track:1gXBi2I04CLJkTQnhNfEJT',
 'spotify:track:7DfFc7a6Rwfi3YQMRbDMau',
 'spotify:track:6TaqooOXAEcijL6G1AWS2K',
 'spotify:track:4FHu9bOzBjZurx89CMa42L',
 'spotify:track:7JXZq0JgG2zTrSOAgY8VMC',
 'spotify:track:1Tt4sE4pXi57mTD1GCzsqm',
 'spotify:track:1Bqxj0aH5KewYHKUg1IdrF',
 'spotify:track:6JG0qhINKVwiHxqN85j7RG',
 'spotify:track:0PJIbOdMs3bd5AT8liULMQ',
 'spotify:track:5NbbXcbCFwOH113niPHDjP',
 'spotify:track:4SYUUlkScpNR1QvPscXf8t',
 'spotify:track:6fTdcGsjxlAD9PSkoPaLMX',
 'spotify:track:7lL3MvFWFFSD25pBz72Agj',
 'spotify:track:62vpWI1CHwFy7tMIcSStl8',
 'spotify:track:2QgNcjwEn0vGmTommfszQd',
 'spotify:track:1c35oVynBuaFs99MNGUOUj',
 'spotify:track:2JrttSXqvhuac7ppgNrnl9',
 'spotify:track:6aYwPh0dchCcdRWHEPUUai',
 'spotify:track:7t0RGhnXGBoVfy3Gakip8B',
 'spotify:track:6GnhWMhgJb7uyiiPEiEkDA',
 'spotify:track:6ficUPpshKKUmKa6SLN7Oq',
 'spotify:track:2IO7yf562c1zLzpanal1DT',
 'spotify:track:17YuXw2ScwLLL1sUrRKhoW',
 'spotify:track:0HDJk7Sz8DpCsiJR1sjhEO',
 'spotify:track:7gDwRzJhLs0fFHaGtMDJVM',
 'spotify:track:13TmKVWEszgLfYyIzVhKkw',
 'spotify:track:75GSFat05fUVklWAPKvVnH',
 'spotify:track:2vTdEBKuXAFi6T0IbRuDff']
```

In [65]:
```
mostSimilar('spotify:track:2vTdEBKuXAFi6T0IbRuDff', 12)
```

Out[65]:
```
[(0.4, 'spotify:track:2TBWfzWL2h1HYSjvQOndyL'),
 (0.3333333333333333, 'spotify:track:2CylUcZGI81nhek3kJQkJ3'),
 (0.3333333333333333, 'spotify:track:5yVeXyRBRYPCimpTyoN0Tb'),
 (0.3333333333333333, 'spotify:track:74A6zH80mhDbjm5Fm49e9d'),
 (0.3333333333333333, 'spotify:track:0uQyNbVmMuplWM00S8dTaS'),
 (0.3333333333333333, 'spotify:track:6is48lQXsZ6hn87nrYANI3'),
 (0.3333333333333333, 'spotify:track:3LgvRAGHfSfq2QzNQLNku8'),
 (0.3333333333333333, 'spotify:track:2niUiaoJDCzNldW0pV66zb'),
 (0.3333333333333333, 'spotify:track:2afr2dQyH6VtOzD8kwS3hl'),
 (0.3333333333333333, 'spotify:track:1y3xRt1NBbBKqi7CHu5Erp'),
 (0.3333333333333333, 'spotify:track:5U1lQBg8zLFZCv9UzU6Noh'),
 (0.3333333333333333, 'spotify:track:7vXKZJy7buchKnncu1meOh')]
```

In [69]:
```
mostSimilarUsers(122001, 10)
```

Out[69]:
```
[(0.08888888888888889, 582772),
 (0.06451612903225806, 995330),
 (0.0625, 122078),
 (0.0625, 308240),
 (0.06153846153846154, 122407),
 (0.061224489795918366, 308952),
 (0.058823529411764705, 308777),
 (0.05555555555555555, 582976),
 (0.05128205128205128, 308920),
 (0.05128205128205128, 995917)]
```

In [76]:
```
playlistsPerTrack['spotify:track:2vTdEBKuXAFi6T0IbRuDff']
```

Out[76]:
```
{122125, 122565, 995551}
```

In [70]:
```
tracksPerPlaylist[582772]
```

Out[70]:
```
{'spotify:track:0PJIbOdMs3bd5AT8liULMQ',
 'spotify:track:0nnRUsgfm4uQUYTCIH9NZy',
 'spotify:track:11KJSRSgaDxqydKYiD2Jew',
 'spotify:track:1ACZpHI5vZ5Ea4xGlkdGWM',
 'spotify:track:1Tt4sE4pXi57mTD1GCzsqm',
 'spotify:track:1UZ25gykR30Oewh3dBRtVZ',
 'spotify:track:282L6SR4Y8Rs0VUgtEy1Zw',
 'spotify:track:2LvRR121MWFmmEGkuV2vQP',
 'spotify:track:40YcuQysJ0KlGQTeGUosTC',
 'spotify:track:4Fv6wNYUixnYkj3Dgfrls8',
 'spotify:track:4K0zMUoKybQpmyb9Y3y8mJ',
 'spotify:track:5DSr1NMci58MEgS20vivSL',
 'spotify:track:5O2Yjlcs5pSpHq1gcA0cwg',
 'spotify:track:5hTpBe8h35rJ67eAWHQsJx',
 'spotify:track:5pfJsMwoRYKampPay8amX0',
 'spotify:track:62vpWI1CHwFy7tMIcSStl8',
 'spotify:track:6F609ICg9Spjrw1epsAnpa',
 'spotify:track:6JG0qhINKVwiHxqN85j7RG',
 'spotify:track:7lQ8MOhq6IN2w8EYcFNSUk',
 'spotify:track:7rdjfrTBMNt3KaaGvSv3YG',
 'spotify:track:7yyRTcZmCiyzzJlNzGC9Ol'}
```

In [53]:
```
def Jaccard(s1, s2):
    numer = len(s1.intersection(s2))
    denom = len(s1.union(s2))
    if denom == 0:
        return 0
    return numer / denom
```

In [284]:
```
def mostSimilar(i, N, tt = None):
    similarities = []
    playlists = playlistsPerTrack[i]
    ct = 0
    if tt is not None:
        p = tt
    else:
        p = playlistsPerTrack
    for i2 in p:
#         i2 = i1['track_uri']
        if i2 == i: continue
        sim = Jaccard(playlists, playlistsPerTrack[i2])
        #sim = Pearson(i, i2) # Could use alternate similarity metrics
        similarities.append((sim,i2))
    similarities.sort(reverse=True, key=lambda x: x[0])
    return similarities[:N]
```

In [245]:
```
def mostSimilarWithPopular(i, N):
    similarities = []
    playlists = playlistsPerTrack[i]
    playlistsTracks = []
    for p in playlists:
        playlistTracks += p['tracks']

    print (playlistsTracks)
    ct = 0
    for i2, _ in mp_tracks.iterrows():
#         i2 = i1['track_uri']
        if i2 == i: continue
        ct += 1
        if ct == 25:
            break
        playlists = playlistsPerTrack[i2]
        playlistsTracks1 = []
        for p in playlists:
            playlistTracks1 += p['tracks']
        sim = Jaccard(playlistTracks, playlistsTracks1)
        #sim = Pearson(i, i2) # Could use alternate similarity metrics
        similarities.append((sim,i2))
    similarities.sort(reverse=True, key=lambda x: x[0])
    return similarities[:N]
```

In [251]:
```
def mostSimilarUsers(u, N):
    similarities = []
    items = tracksPerPlaylist[u]
    for u2 in tracksPerPlaylist:
        if u2 == u: continue
        sim = Jaccard(items, tracksPerPlaylist[u2])
        #sim = Pearson(i, i2) # Could use alternate similarity metrics
        similarities.append((sim,u2))
    similarities.sort(reverse=True, key=lambda x: x[0])
    return similarities[:N]
```

In [277]:
```
tracksPerPlaylist[440143]
```

Out[277]:
```
{'spotify:track:0BweE3lWBMXRPWWLtLV5z8',
 'spotify:track:0Fx3R5YWHVpDEtO9z7MI16',
 'spotify:track:0OL5HxUb75j3bLX61VEN5M',
 'spotify:track:0UvqEDeLKHSt7r9wh7Bk0e',
 'spotify:track:0ed2A7qGzPRznZWblCqLzw',
 'spotify:track:0rvxJokXx4RTidsm9C9JH0',
 'spotify:track:1Gsv8f8KmowkF5BnfMIGKy',
 'spotify:track:1OFbVwytaZkVxcbzlKjevB',
 'spotify:track:1QQfbfMfyzNZhT5fg3RpuS',
 'spotify:track:1mB5iZe1OrN9gYZtLOklI6',
 'spotify:track:1zjcaUqHF1ChT65klNMtwF',
 'spotify:track:21dOjdraFZffs2lnQObaiZ',
 'spotify:track:22AbXxQbMdVqEz7xJjhccG',
 'spotify:track:260F1LrhRMiuXOZIaypMZd',
 'spotify:track:286bpRdcpueYPc7gLIvrzU',
 'spotify:track:3LYv1fJdCLMBvsqDyzHrCT',
 'spotify:track:3QhU6WcZYm9nl3Z0ARzdlu',
 'spotify:track:3kfrej04kCc2958ERutmsX',
 'spotify:track:3xbKyhn7Bk0XDVecWF793N',
 'spotify:track:4GOThCLUMQiqye7ejz2eyh',
 'spotify:track:4OsZ1vrenrtSbqLJxOceKl',
 'spotify:track:4TyCnstYu6LGrjka5WW6ft',
 'spotify:track:4ZzqePQcoVCQKd4eH2q6xp',
 'spotify:track:4a9jp87IPyuL09hjehzDGE',
 'spotify:track:4d9dLobDQ4BcfNbOnSPHQC',
 'spotify:track:4efoEY8iDBzUqitjmNDhpN',
 'spotify:track:4gU4mrhf4qBBlPNeBhHe8R',
 'spotify:track:52EGIALITgYJkWVlq42VfK',
 'spotify:track:5DKFbJteHTFm0gT96lhJ2E',
 'spotify:track:5xkgdScv3FmHNYm0o6AZVe',
 'spotify:track:61dXuN3VyqGOyOeMfJXIPh',
 'spotify:track:6M5cQCyieE1Q8YQydVLzxL',
 'spotify:track:6kphCDUPSapbKSoQaemQjo',
 'spotify:track:6pL7WLNdWAtDS9c2Xz0X0u',
 'spotify:track:76iLedMTAC15Nl4opm2bjb',
 'spotify:track:78IxKAvzvPUxp30Skp28Qy',
 'spotify:track:7K1QvbTr4KqK8Np5SpFC6S'}
```

In [130]:
```
ms = []
for t in tqdm(tracksPerPlaylist[122001]):
    ms.extend(mostSimilar(t, 2))

ms.sort(reverse=True)
```

```
---------------------------------------------------------------------
------
TypeError                                 Traceback (most recent call
last)
/var/folders/k1/cn4r859n1d3gq3qrhj73qzsh0000gn/T/ipykernel_2132/28515
14750.py in <module>
      1 ms = []
----> 2 for t in tqdm(tracksPerPlaylist[122001]):
      3     ms.extend(mostSimilar(t, 2))
      4
      5 ms.sort(reverse=True)

TypeError: 'module' object is not callable
```

In [244]:
```
mostSimilar('spotify:track:0HDJk7Sz8DpCsiJR1sjhEO', 2)
```

Out[244]:
```
[(0.2857142857142857, 'spotify:track:1yCVsVH2hQ72SxNI8QTDaB'),
 (0.16666666666666666, 'spotify:track:2JrttSXqvhuac7ppgNrnl9')]
```

In [75]:
```
tracksPerPlaylist[122001]
```

Out[75]:
```
{'spotify:track:0HDJk7Sz8DpCsiJR1sjhEO',
 'spotify:track:0PJIbOdMs3bd5AT8liULMQ',
 'spotify:track:13TmKVWEszgLfYyIzVhKkw',
 'spotify:track:17YuXw2ScwLLL1sUrRKhoW',
 'spotify:track:1Bqxj0aH5KewYHKUg1IdrF',
 'spotify:track:1Tt4sE4pXi57mTD1GCzsqm',
 'spotify:track:1c35oVynBuaFs99MNGUOUj',
 'spotify:track:1gXBi2I04CLJkTQnhNfEJT',
 'spotify:track:1wZqJM5FGDEl3FjHDxDyQd',
 'spotify:track:2IO7yf562c1zLzpanal1DT',
 'spotify:track:2JrttSXqvhuac7ppgNrnl9',
 'spotify:track:2QgNcjwEn0vGmTommfszQd',
 'spotify:track:4FHu9bOzBjZurx89CMa42L',
 'spotify:track:4SYUUlkScpNR1QvPscXf8t',
 'spotify:track:5NbbXcbCFwOH113niPHDjP',
 'spotify:track:62vpWI1CHwFy7tMIcSStl8',
 'spotify:track:6GnhWMhgJb7uyiiPEiEkDA',
 'spotify:track:6JG0qhINKVwiHxqN85j7RG',
 'spotify:track:6TaqooOXAEcijL6G1AWS2K',
 'spotify:track:6aYwPh0dchCcdRWHEPUUai',
 'spotify:track:6fTdcGsjxlAD9PSkoPaLMX',
 'spotify:track:6ficUPpshKKUmKa6SLN7Oq',
 'spotify:track:75GSFat05fUVklWAPKvVnH',
 'spotify:track:7DfFc7a6Rwfi3YQMRbDMau',
 'spotify:track:7JXZq0JgG2zTrSOAgY8VMC',
 'spotify:track:7gDwRzJhLs0fFHaGtMDJVM',
 'spotify:track:7lL3MvFWFFSD25pBz72Agj',
 'spotify:track:7t0RGhnXGBoVfy3Gakip8B'}
```

In [110]:
```
ms.sort(reverse=True)
ms
```

Out[110]:
```
[(0.12048192771084337, 'spotify:track:1xznGGDReH1oQq0xzbwXa3'),
 (0.08823529411764706, 'spotify:track:343YBumqHu19cGoGARUTsd'),
 (0.08333333333333333, 'spotify:track:5hTpBe8h35rJ67eAWHQsJx'),
 (0.07792207792207792, 'spotify:track:5tz69p7tJuGPeMGwNTxYuV'),
 (0.07407407407407407, 'spotify:track:7yyRTcZmCiyzzJlNzGC9Ol'),
 (0.06666666666666667, 'spotify:track:343YBumqHu19cGoGARUTsd'),
 (0.06578947368421052, 'spotify:track:152lZdxL1OR0ZMW6KquMif'),
 (0.06153846153846154, 'spotify:track:0v9Wz8o0BT8DU38R4ddjeH'),
 (0.058823529411764705, 'spotify:track:343YBumqHu19cGoGARUTsd'),
 (0.05, 'spotify:track:5hTpBe8h35rJ67eAWHQsJx'),
 (0.05, 'spotify:track:5hTpBe8h35rJ67eAWHQsJx'),
 (0.04395604395604396, 'spotify:track:152lZdxL1OR0ZMW6KquMif'),
 (0.04285714285714286, 'spotify:track:7yyRTcZmCiyzzJlNzGC9Ol'),
 (0.03773584905660377, 'spotify:track:5hTpBe8h35rJ67eAWHQsJx'),
 (0.03508771929824561, 'spotify:track:0v9Wz8o0BT8DU38R4ddjeH'),
 (0.0338983050847457, 'spotify:track:7GX5flRQZVHRAGd6B4TmDO'),
```

```
 (0.03389830508474576, 'spotify:track:7GX5flRQZVHRAGd6B4TmDO'),
 (0.03333333333333333, 'spotify:track:5tz69p7tJuGPeMGwNTxYuV'),
 (0.03333333333333333, 'spotify:track:1xznGGDReH1oQq0xzbwXa3'),
 (0.03225806451612903, 'spotify:track:1xznGGDReH1oQq0xzbwXa3'),
 (0.029850746268656716, 'spotify:track:152lZdxL1OR0ZMW6KquMif'),
 (0.02857142857142857, 'spotify:track:0v9Wz8o0BT8DU38R4ddjeH'),
 (0.028169014084507043, 'spotify:track:0QsvXIfqM0zZoerQfsI9lm'),
 (0.025974025974025976, 'spotify:track:32OlwWuMpZ6b0aN2RZOeMS'),
 (0.02, 'spotify:track:1xznGGDReH1oQq0xzbwXa3'),
 (0.0196078431372549, 'spotify:track:32OlwWuMpZ6b0aN2RZOeMS'),
 (0.0196078431372549, 'spotify:track:2d8JP84HNLKhmd6IYOoupQ'),
 (0.019230769230769232, 'spotify:track:7yyRTcZmCiyzzJlNzGC9Ol'),
 (0.019230769230769232, 'spotify:track:4Km5HrUvYTaSUfiSGPJeQR'),
 (0.018867924528301886, 'spotify:track:2d8JP84HNLKhmd6IYOoupQ'),
 (0.018518518518518517, 'spotify:track:5tz69p7tJuGPeMGwNTxYuV'),
 (0.01818181818181818, 'spotify:track:0v9Wz8o0BT8DU38R4ddjeH'),
 (0.017543859649122806, 'spotify:track:2d8JP84HNLKhmd6IYOoupQ'),
 (0.017241379310344827, 'spotify:track:0QsvXIfqM0zZoerQfsI9lm'),
 (0.016666666666666666, 'spotify:track:343YBumqHu19cGoGARUTsd'),
 (0.016666666666666666, 'spotify:track:0QsvXIfqM0zZoerQfsI9lm'),
 (0.01639344262295082, 'spotify:track:0KKkJNfGyhkQ5aFogxQAPU'),
 (0.015384615384615385, 'spotify:track:7GX5flRQZVHRAGd6B4TmDO'),
 (0.015384615384615385, 'spotify:track:6O6M7pJLABmfBRoGZMu76Y'),
 (0.014492753623188406, 'spotify:track:0SGkqnVQo9KPytSri1H6cF'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7KXjTSCq5nL1LoYtL7XAwS'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw'),
 (0.0, 'spotify:track:7BKLCZ1jbUBVqRi2FVlTVw')]
```

In [ ]:

In [105]:
```
for i, t in c.iterrows():
    print (i)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
/var/folders/k1/cn4r859n1d3gq3qrhj73qzsh0000gn/T/ipykernel_2132/2113491647.py in <module>
----> 1 for i, t in c.iterrows().head(2):
      2     print (i)

AttributeError: 'generator' object has no attribute 'head'
```

In [ ]:
```
```

Get test data

In [191]:
```
all_playlists_test = pd.DataFrame()
all_tracks_test = pd.DataFrame()
L = []

for fileName in ['mpd.slice.440000-440999.json']:
    f = open(fileName)
    data = json.load(f)
    l, playlist_df, tracks_df = loadPlaylistData(data)
    # print(playlist_df)
    L.append(l)
    all_playlists_test = pd.concat([all_playlists_test, playlist_df])
    all_tracks_test = pd.concat([all_tracks_test, tracks_df])
```

In [593]:
```
testData = []
y_test = []
c = 0


for idx, playlist in all_playlists_test.iterrows():
    c += 1
#     if c == 11:
#         break
    tempPlaylist = copy.deepcopy(playlist)
    tempTracks = []

#     print (idx)
    for i in range(len(playlist['tracks']) - 10):
        tempTracks.append(playlist['tracks'][i])
        tr = all_tracks_test.loc[playlist['tracks'][i]]
        tracksPerArtist[tr['artist_uri']].add(playlist['tracks'][i])
        playlistsPerTrack[playlist['tracks'][i]].add(idx)
        artistsPerTrack[playlist['tracks'][i]].add(tr['artist_uri'])
        tracksPerPlaylist[idx].add(playlist['tracks'][i])

    withHeldTracks = []
    withHeldArtists = []
    for i in range(len(playlist['tracks']) - 10, len(playlist['tracks'
        tr = all_tracks_test.loc[playlist['tracks'][i]]
        playlistsPerTrack[playlist['tracks'][i]].add(idx)
        tracksPerPlaylist[idx].add(playlist['tracks'][i])
        artistsPerTrack[playlist['tracks'][i]].add(tr['artist_uri'])
        tracksPerArtist[tr['artist_uri']].add(playlist['tracks'][i])

        tr = all_tracks_test.loc[playlist['tracks'][i]]
        withHeldTracks.append(playlist['tracks'][i])
        withHeldArtists.append(tr['artist_uri'])

    tempPlaylist['tracks'] = tempTracks
    tempPlaylist['withHeldTracks'] = withHeldTracks
    tempPlaylist['withHeldArtists'] = withHeldArtists

    testData.append(tempPlaylist)

    p = copy.deepcopy(tempPlaylist)
    t = [playlist['tracks'][-1]]
    p['tracks'] = t
    y_test.append(playlist['tracks'][-1])
```

In [429]:
```
testData[0]
```

Out[429]:
```
name                                                          mood
collaborative                                                false
pid                                                         440000
modified_at                                             1506384000
num_tracks                                                      23
num_albums                                                      14
num_followers                                                    1
tracks            [spotify:track:21bl9jnt8gKltnNYpOPbhw, spotify...
num_edits                                                       16
duration_ms                                                5030548
num_artists                                                     11
description                                                    NaN
withHeldTracks    [spotify:track:75ZvA4QfFiZvzhj2xkaWAh, spotify...
withHeldArtists   [spotify:artist:246dkjvS1zLTtiykXe5h60, spotif...
Name: 440000, dtype: object
```

In [481]:
```
tracksPerArtist
```

```
                'spotify:artist:5tKXB9uuebKE34yowVaU3C': {'spotify:track
:1TEXhp3NMV9U3xvWFZy5Xq',
                 'spotify:track:4E5P1XyAFtrjpiIxkydly4',
                 'spotify:track:7ElF5zxOwYP4qVSWVvse3W'},
                'spotify:artist:3nFkdlSjzX9mRTtwJOzDYB': {'spotify:track
:0ESJlaM8CE1jRWaNtwSNj8',
                 'spotify:track:1mU0es3QUBRAxKkYvTANpG',
                 'spotify:track:25CA8QVJQrh5R05UUCaODM',
                 'spotify:track:2KpCpk6HjXXLb7nnXoXA5O',
                 'spotify:track:2LvRR121MWFmmEGkuV2vQP',
                 'spotify:track:3aHkbJXFYpPrup2K8wvDOV',
                 'spotify:track:3lPAHV0b53zgRRm9LBgnTY',
                 'spotify:track:3rbNV2GI8Vtd8byhUtXZID',
                 'spotify:track:5IVCuswUfdYjl2eQ2FHqUd',
                 'spotify:track:5sNESr6pQfIhL3krM8CtZn',
                 'spotify:track:6MmT9hje2krkn1twcmtBYu',
                 'spotify:track:722tgOgdIbNe3BEyLnejw4',
                 'spotify:track:7zBQRGpYImAdIZc97FNj3V'},
                'spotify:artist:0hCNtLu0JehylgoiP8L4Gh': {'spotify:track
```

In [276]:
```
all_playlists_test.loc[440143]
```

Out[276]:
```
name                                                          reggae
collaborative                                                  false
pid                                                           440143
modified_at                                                1502582400
num_tracks                                                         38
num_albums                                                         21
num_followers                                                       1
tracks           [spotify:track:4a9jp87IPyuL09hjehzDGE, spotify...
num_edits                                                          11
duration_ms                                                   8635520
num_artists                                                        12
description                                 music 4 the mind and soul
Name: 440143, dtype: object
```

In [327]:
```
y_test
```

```
 'spotify:track:2FjXXVgsDQ8VdgmuWEYDun',
 'spotify:track:768YahmAf7ZnbUeWPFp72u',
 'spotify:track:1YrnDTqvcnUKxAIeXyaEmU',
 'spotify:track:6UjfByV1lDLW0SOVQA4NAi',
 'spotify:track:3HU4vFJPelZfj0xyJDVALk',
 'spotify:track:2rzBvHM9h36Tpdj7Jdajka',
 'spotify:track:43GS3mtezoIFiuIZCLLiDY',
 'spotify:track:4gmmRb6bZJffOOiww1JGTO',
 'spotify:track:0cG9ocYqYOIVtgiWTmTe3g',
 'spotify:track:0vdR0X9kDRDPWZMQ4gHo48',
 'spotify:track:6f5qWGKdQ1uIe27Hbmccxa',
 'spotify:track:3WKg25vrbjJlkhsgl2W4p3',
 'spotify:track:1KcXwiSdMEyH2N2vs7Htbz',
 'spotify:track:2Y48Q7HryWdPJZypJotjlJ',
 'spotify:track:3nIlXRlu5AuZhAsWJgOipm',
 'spotify:track:2uDngplEd0RW5hqcAur5uY',
 'spotify:track:0IzIW2MSFdllQ7zFLbv1uS',
 'spotify:track:6kfFx81E5OPb6C3UHz5PTp',
 'spotify:track:4xyNG8vZR3g8FnphVkbjRo',
 'spotify:track:7x6cBmHXwdS1EGwVx4sjlu',
```

In [482]:
```
artistsPerTrack
```

Out[482]:
```
defaultdict(set,
            {'spotify:track:21bl9jnt8gKltnNYpOPbhw': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:4wXvqEHxo3HwLs6mcqU4bb': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:49VzL5oBsUICN7U34T98b9': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:2yTTfOaptQVJiocgjoQFfi': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:4cgWdfBPK7C3KYNeCtl11j': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:4XnajLeZt02RnYg4k8kXpm': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:4SzyebNrhvPJ8y4r1D1PXD': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:2bxG0N7YsdkIBfuLsE0J8i': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
             'spotify:track:07Qonra5xGHyjndj0gtOHy': {'spotify:artist
:7mDU6nMUJnOSY2Hkjz5oqM'},
```

In [448]:
```
mp_tracks[:10]
```

Out[448]:

track_uri	pos	artist_name	tra
spotify:track:7BKLCZ1jbUBVqRi2FVlTVw	24	The Chainsmokers	spotify:track:7BKLCZ1jbUBVqRi2I
spotify:track:7KXjTSCq5nL1LoYtL7XAwS	36	Kendrick Lamar	spotify:track:7KXjTSCq5nL1LoYtL
spotify:track:152lZdxL1OR0ZMW6KquMif	28	Khalid	spotify:track:152lZdxL1OR0ZMW6
spotify:track:3DXncPQOG4VBw3QHh3S817	1	DJ Khaled	spotify:track:3DXncPQOG4VBw3QHh
spotify:track:0SGkqnVQo9KPytSri1H6cF	21	Big Sean	spotify:track:0SGkqnVQo9KPytSr
spotify:track:2EEeOnHehOozLq4aS0n6SL	29	KYLE	spotify:track:2EEeOnHehOozLq4aS
spotify:track:6O6M7pJLABmfBRoGZMu76Y	0	The Chainsmokers	spotify:track:6O6M7pJLABmfBRoGZ
spotify:track:7GX5flRQZVHRAGd6B4TmDO	5	Lil Uzi Vert	spotify:track:7GX5flRQZVHRAGd6B4
spotify:track:3a1lNhkSLSkpJE4MSHpDu9	8	Post Malone	spotify:track:3a1lNhkSLSkpJE4MS
spotify:track:0KKkJNfGyhkQ5aFogxQAPU	37	Bruno Mars	spotify:track:0KKkJNfGyhkQ5aFogx

In [489]:
```
all_tracks.loc['spotify:track:75ZvA4QfFiZvzhj2xkaWAh']
```

Out[489]:
```
pos                                                38
artist_name                                 Post Malone
track_uri       spotify:track:75ZvA4QfFiZvzhj2xkaWAh
artist_uri     spotify:artist:246dkjvS1zLTtiykXe5h60
track_name                                I Fall Apart
album_uri       spotify:album:5s0rmjP8XOPhP6HhqOhuyC
duration_ms                                    223346
album_name                                     Stoney
popularity                                         21
Name: spotify:track:75ZvA4QfFiZvzhj2xkaWAh, dtype: object
```

In [ ]:
```

```

In [487]:
```
tracksPerArtist['spotify:artist:0YMDurwdAVosp0voDcFdFG']
```

Out[487]:
```
{'spotify:track:75ZvA4QfFiZvzhj2xkaWAh'}
```

In [490]:
```
artistsPerTrack
```

```
:25hbSOMmbhgqvonjC876UJ'},
             'spotify:track:0e3aRkhcCdkYN62p2PFfD3': {'spotify:artist
:4j0qbafzqnGf4ARF8tY4VL'},
             'spotify:track:6M5cQCyieE1Q8YQydVLzxL': {'spotify:artist
:2QsynagSdAqZj3U9HgDzjD'},
             'spotify:track:4efoEY8iDBzUqitjmNDhpN': {'spotify:artist
:2QsynagSdAqZj3U9HgDzjD'},
             'spotify:track:61dXuN3VyqGOyOeMfJXIPh': {'spotify:artist
:2QsynagSdAqZj3U9HgDzjD'},
             'spotify:track:4OsZ1vrenrtSbqLJxOceKl': {'spotify:artist
:2QsynagSdAqZj3U9HgDzjD'},
             'spotify:track:1QQfbfMfyzNZhT5fg3RpuS': {'spotify:artist
:2QsynagSdAqZj3U9HgDzjD'},
             'spotify:track:21dOjdraFZffs2lnQObaiZ': {'spotify:artist
:2QsynagSdAqZj3U9HgDzjD'},
             'spotify:track:7CLM0jEXuHeGWBLJgLwYOQ': {'spotify:artist
:5Ayl2bJtN5mdCsxZoxs9n1'},
             'spotify:track:6dViIgkYIRccNImPln58rS': {'spotify:artist
:5EvFsr3kj42KNv97ZEnqij'},
             'spotify:track:0YvwiDvFudcaHG74NuWISv': {'spotify:artist
```

In [517]:
```
getTracksForArtists(['spotify:artist:3TVXtAsR1Inumwj472S9r4'])
```

```
['spotify:artist:3TVXtAsR1Inumwj472S9r4']
Q {'spotify:track:5mPSyjLatqB00IkPqRlbTE', 'spotify:track:1pF5hkdEGJv
8xzLTvhy3PF', 'spotify:track:0V4l4GQhgnWQGtCWpvA7va', 'spotify:track:
2AGottAzfC8bHzF7kEJ3Wa', 'spotify:track:3NxAG2ni1lLa8RKL6a0INc', 'spo
tify:track:7MjSipTto9QljYzZnloXOn', 'spotify:track:27GmP9AWRs744SzKcp
JsTZ', 'spotify:track:6BdgtqiV3oXNqBikezwdvC', 'spotify:track:4pc01CA
hGKx15PY23uPjHp', 'spotify:track:1xznGGDReH1oQq0xzbwXa3', 'spotify:tr
ack:4Kz4RdRCceaA9VgTqBhBfa', 'spotify:track:343YBumqHu19cGoGARUTsd',
'spotify:track:4ckuS4Nj4FZ7i3Def3Br8W', 'spotify:track:2fkeWbM6iqTw7o
GHTYm2lw', 'spotify:track:7yfg0Eer6UZZt5tZ1XdsWz', 'spotify:track:6V2
D8Lls36APk0THDjBDfE', 'spotify:track:2bjwRfXMk4uRgOD9IBYl9h', 'spotif
y:track:7sBwAWyXfiIgrYQ8BaJESH', 'spotify:track:6UjfByV1lDLW0SOVQA4NA
i', 'spotify:track:6Z01gUquJsjJC67uNWm6P0', 'spotify:track:2XlHu0Hcuj
BCkWMdIAvrqt', 'spotify:track:2FbGlEPAjNhWvrVvlentVq', 'spotify:track
:31Q9ZTF9x81BDonlObCbvP', 'spotify:track:5EnYT6F7wEcdege6mDHEfO', 'sp
otify:track:0jF2AdhsalO1L7KkhK4LE5', 'spotify:track:3lSR267IJfT54p0Gf
uw7mi', 'spotify:track:7bJ4mu7MHa3rHiNyKjOoSl', 'spotify:track:7jslhI
iELQkgW9IHeYNOWE', 'spotify:track:047fCsbO4NdmwCBn8pcUXl', 'spotify:t
rack:6CfrYuD3YRDYdYvH9jNtXY', 'spotify:track:7hDc8b7IXETo14hHIHdnhd',
'spotify:track:6MbH1QiphMCPTqVEVC7UYi', 'spotify:track:4BhGTc3Cgay2U1
QcTS7vQe', 'spotify:track:11KJSRSgaDxqydKYiD2Jew', 'spotify:track:2qL
cJOLrh6Djda4uLbldSA', 'spotify:track:3ptQ2qKjiGOIW1USCFXVtT', 'spotif
y:track:3fyMH1t6UPeR5croea9PrR', 'spotify:track:10VBBaul4zVD0reteuIHM
2', 'spotify:track:76gUmNLXGQVOsGhfcshkFP', 'spotify:track:4gowy3WT6D
1yhMLgRBlf9C', 'spotify:track:1ID1QFSNNxi0hiZCNcwjUC', 'spotify:track
```

```
otify:track:2ZRJRe82aZaVhOKKlbJr4v', 'spotify:track:1C7KSXR2GVxknex6I

4ANco', 'spotify:track:124NFj84ppZ5pAxTuVQYCQ', 'spotify:track:13e6f8
t7RKXuxZ0JdaaJRG', 'spotify:track:7JXZq0JgG2zTrSOAgY8VMC', 'spotify:t
rack:5NFYuqu8V6QXc6mzcLImd6', 'spotify:track:4cRBqWBjuccCowYVHFlXK6',
'spotify:track:1HDaPtZuixue2q6VGNRdVO', 'spotify:track:4eSGSqP2TZvvX0
kadZZttM', 'spotify:track:6F609ICg9Spjrw1epsAnpa', 'spotify:track:6Lx
Se8YmdPxy095Ux6znaQ', 'spotify:track:433P7tDcIAi6NLnf4Sh6tI', 'spotif
y:track:05KOgYg8PGeJyyWBPi5ja8', 'spotify:track:6jdOi5U5LBzQrc4c1VT98
3', 'spotify:track:3cjF2OFRmip8spwZYQRKxP', 'spotify:track:76kyKtPLsF
bQkdQ86QrkF4', 'spotify:track:7udsBKuqnJ5csWTAkR0vEI', 'spotify:track
:79XrkTOfV1AqySNjVlygpW', 'spotify:track:2YaDRtIlQiZ5WDDB2YuEOC', 'sp
otify:track:1DmnEYXa4WfbdhAPwNzgD8', 'spotify:track:0m1KYWlT6LhFRBDVq
9UNx4', 'spotify:track:2jTujnt0y344ai1rNOywgr', 'spotify:track:2FBUoW
kIuQXwayw2RNo5l6'}
```

Out[517]:
```
['spotify:track:5mPSyjLatqB00IkPqRlbTE',
 'spotify:track:1pF5hkdEGJv8xzLTvhy3PF',
 'spotify:track:0V4l4GQhgnWQGtCWpvA7va',
 'spotify:track:2AGottAzfC8bHzF7kEJ3Wa',
 'spotify:track:3NxAG2ni1lLa8RKL6a0INc',
 'spotify:track:7MjSipTto9QljYzZnloXOn',
 'spotify:track:27GmP9AWRs744SzKcpJsTZ',
 'spotify:track:6BdgtqiV3oXNqBikezwdvC',
 'spotify:track:4pc01CAhGKx15PY23uPjHp',
 'spotify:track:1xznGGDReH1oQq0xzbwXa3',
 'spotify:track:4Kz4RdRCceaA9VgTqBhBfa',
 'spotify:track:343YBumqHu19cGoGARUTsd',
 'spotify:track:4ckuS4Nj4FZ7i3Def3Br8W',
 'spotify:track:2fkeWbM6iqTw7oGHTYm2lw',
 'spotify:track:7yfg0Eer6UZZt5tZ1XdsWz',
 'spotify:track:6V2D8Lls36APk0THDjBDfE',
 'spotify:track:2bjwRfXMk4uRgOD9IBYl9h',
 'spotify:track:7sBwAWyXfiIgrYQ8BaJESH',
 'spotify:track:6UjfByV1lDLW0SOVQA4NAi',
 'spotify:track:6Z01gUquJsjJC67uNWm6P0',
 'spotify:track:2XlHu0HcujBCkWMdIAvrqt',
 'spotify:track:2FbGlEPAjNhWvrVvlentVq',
 'spotify:track:31Q9ZTF9x81BDonlObCbvP',
 'spotify:track:5EnYT6F7wEcdege6mDHEfO',
 'spotify:track:0jF2AdhsalO1L7KkhK4LE5',
 'spotify:track:3lSR267IJfT54p0Gfuw7mi',
 'spotify:track:7bJ4mu7MHa3rHiNyKjOoSl',

 'spotify:track:7jslhIiELQkgW9IHeYNOWE',
 'spotify:track:047fCsbO4NdmwCBn8pcUXl',
 'spotify:track:6CfrYuD3YRDYdYvH9jNtXY',
 'spotify:track:7hDc8b7IXETo14hHIHdnhd',
 'spotify:track:6MbH1QiphMCPTqVEVC7UYi',
 'spotify:track:4BhGTc3Cgay2U1QcTS7vQe',
 'spotify:track:11KJSRSqaDxqydKYiD2Jew',
```

```
 'spotify:track:2qLcJOLrh6Djda4uLbldSA',
 'spotify:track:3ptQ2qKjiGOIW1USCFXVtT',
 'spotify:track:3fyMH1t6UPeR5croea9PrR',
 'spotify:track:10VBBaul4zVD0reteuIHM2',
 'spotify:track:76gUmNLXGQVOsGhfcshkFP',
 'spotify:track:4gowy3WT6D1yhMLgRBlf9C',
 'spotify:track:1ID1QFSNNxi0hiZCNcwjUC',
 'spotify:track:3aJkV6DUTSCqOwVwaBDG9B',
 'spotify:track:4kNvYhyl8R6m1vykVkcuBu',
 'spotify:track:2ZRJRe82aZaVhOKKlbJr4v',
 'spotify:track:1C7KSXR2GVxknex6I4ANco',
 'spotify:track:124NFj84ppZ5pAxTuVQYCQ',
 'spotify:track:13e6f8t7RKXuxZ0JdaaJRG',
 'spotify:track:7JXZq0JgG2zTrSOAgY8VMC',
 'spotify:track:5NFYuqu8V6QXc6mzcLImd6',
 'spotify:track:4cRBqWBjuccCowYVHFlXK6',
 'spotify:track:1HDaPtZuixue2q6VGNRdVO',
 'spotify:track:4eSGSqP2TZvvX0kadZZttM',
 'spotify:track:6F609ICg9Spjrw1epsAnpa',
 'spotify:track:6LxSe8YmdPxy095Ux6znaQ',
 'spotify:track:433P7tDcIAi6NLnf4Sh6tI',
 'spotify:track:05KOgYg8PGeJyyWBPi5ja8',
 'spotify:track:6jdOi5U5LBzQrc4c1VT983',
 'spotify:track:3cjF2OFRmip8spwZYQRKxP',
 'spotify:track:76kyKtPLsFbQkdQ86QrkF4',
 'spotify:track:7udsBKuqnJ5csWTAkR0vEI',
 'spotify:track:79XrkTOfV1AqySNjVlygpW',
 'spotify:track:2YaDRtIlQiZ5WDDB2YuEOC',
 'spotify:track:1DmnEYXa4WfbdhAPwNzgD8',
 'spotify:track:0m1KYWlT6LhFRBDVq9UNx4',
 'spotify:track:2jTujnt0y344ai1rNOywgr',
 'spotify:track:2FBUoWkIuQXwayw2RNo5l6']
```

In [520]:
```
def getArtistsForTracks(tracks):
#   print ("TTT")

  c = [list(artistsPerTrack[track])[0] for track in tracks]
#   print (c)
  return c

def getTracksForArtists(artists):
#   print ("TTT")
  ans = []
#   print (artists)
  for artist in artists:
#     print ("Q", tracksPerArtist[artist])
    ans.extend(list(tracksPerArtist[artist]))
#   c = [list(tracksPerArtist[artist])[0] for artist in artists]
#   print (c)
  return ans
```

In [556]:
```
def getMostPopularSongs(n, seed_playlist):
#   songs_with_freq_list = []
#   for song in track_frequency:
#     songs_with_freq_list.append((track_frequency[song], song))
  tr = mp_tracks['track_uri']
  res = [t for t in tr if t not in seed_playlist]
  return res[:n]

def getMostPopularArtists(n):
#   songs_with_freq_list = []
#   for song in track_frequency:
#     songs_with_freq_list.append((track_frequency[song], song))
  tr = list(mp_artists.index)
#   print (n, tr[:2])
  return tr[:n]
```

In [467]:
```
def R_precision(predicted_tracks, with_held_tracks):
  # print(len(set(with_held_tracks)))
#   print (type(predicted_tracks), type(with_held_tracks))
#   print (set(predicted_tracks))
#   print (set(with_held_tracks))
  intersection = len(set(predicted_tracks).intersection(set(with_held_
  return intersection / len(set(with_held_tracks))

def NDCG(truth, pred, K):
  pred_copy = copy.deepcopy(pred)
  if len(pred) > 0:
    rel_truth = np.zeros((1, max(len(pred), len(truth))))
    rel_pred = np.zeros((1, max(len(pred), len(truth))))
    count_t = 0
    for t in truth:
      if t in pred_copy:
        idx = np.where(np.array(pred_copy) == t)[0]
        pred_copy[idx[0]] = 'NAN'
        rel_pred[0, idx[0]]=1
        rel_truth[0, count_t]=1
      count_t+=1
    return ndcg_score(rel_truth, rel_pred, k=K)
  else:
    return 0
```

In [502]:
```
mp_artists = e
```

In [566]:
```
def runPopularArtistModel(topKgrid):
  for topK in topKgrid:
    tracks_r_precision = []
    artists_r_precision = []
    ndcg_artists = []
    ndcg_tracks = []
    for row in testData:
      res = getMostPopularArtists(topK)#getMostPopularSongs(topK, row[
#       print ("Q", res)
      tracks_of_pred = getTracksForArtists(res)
#       print (res, len(tracks_of_pred), len(row['withHeldTracks']))
      tracks_r_precision.append(R_precision(tracks_of_pred, row['withH
#       print ("artists: ", artists_of_pred)
      artists_r_precision.append(R_precision(res, row['withHeldArtists
      ndcg_artists.append(NDCG(row['withHeldArtists'], res, topK))
      ndcg_tracks.append(NDCG(row['withHeldTracks'], tracks_of_pred, t

    print (len(artists_r_precision), len(tracks_r_precision), len(ndcg
    print('K=',topK, ' ,RP(Artist)=',sum(artists_r_precision)/len(arti
    print('K=',topK, ' ,RP(Track)=',sum(tracks_r_precision)/len(tracks
    print('K=',topK, ' ,NDCG(Artist)=',sum(ndcg_artists)/len(ndcg_arti
    print('K=',topK, ' ,NDCG(Track)=',sum(ndcg_tracks)/len(ndcg_tracks

    # print('K=',topK, ' ,NDCG(Artist)=',NDCG(row['with_held_artists']
    print('=====================================================')
```

In [568]:
```
runPopularArtistModel([10, 20, 30, 40, 50, 60, 70, 100, 200, 300, 400,
```

```
540 540 540
K= 10  ,RP(Artist)= 0.03980305702527923
K= 10  ,RP(Track)= 0.04340020576131689
K= 10  ,NDCG(Artist)= 0.11024329926035752
K= 10  ,NDCG(Track)= 0.0036301386129673117
=====================================================
540 540 540
K= 20  ,RP(Artist)= 0.06665343915343913
K= 20  ,RP(Track)= 0.0712551440329219
K= 20  ,NDCG(Artist)= 0.13053343433211126
K= 20  ,NDCG(Track)= 0.005268287504904239
=====================================================
540 540 540
K= 30  ,RP(Artist)= 0.08519253380364507
K= 30  ,RP(Track)= 0.09228395061728416
K= 30  ,NDCG(Artist)= 0.14062690317942897
K= 30  ,NDCG(Track)= 0.006513759411015326
=====================================================
540 540 540
K= 40  ,RP(Artist)= 0.10380805408583198
K= 40  ,RP(Track)= 0.11054012345679032
```

```
K= 40  ,NDCG(Artist)= 0.1472716523733647
K= 40  ,NDCG(Track)= 0.006999351163529331
==========================================================
540 540 540
K= 50  ,RP(Artist)= 0.12159391534391546
K= 50  ,RP(Track)= 0.12935185185185197
K= 50  ,NDCG(Artist)= 0.15234439096438987
K= 50  ,NDCG(Track)= 0.008228590089644109
==========================================================
540 540 540
K= 60  ,RP(Artist)= 0.13729570840681968
K= 60  ,RP(Track)= 0.14624485596707834
K= 60  ,NDCG(Artist)= 0.15660100774562935
K= 60  ,NDCG(Track)= 0.008678857497326728
==========================================================
540 540 540
K= 70  ,RP(Artist)= 0.15267783656672568
K= 70  ,RP(Track)= 0.1627263374485597
K= 70  ,NDCG(Artist)= 0.16235086474786722
K= 70  ,NDCG(Track)= 0.009291525991622175
==========================================================
540 540 540
K= 100  ,RP(Artist)= 0.19440696649030006
K= 100  ,RP(Track)= 0.20439814814814825
K= 100  ,NDCG(Artist)= 0.17513656661348273
K= 100  ,NDCG(Track)= 0.010694107817855757
==========================================================
540 540 540
K= 200  ,RP(Artist)= 0.2834795708406819
K= 200  ,RP(Track)= 0.29313271604938235
K= 200  ,NDCG(Artist)= 0.192781068960854
K= 200  ,NDCG(Track)= 0.014685900812400659
==========================================================
540 540 540
K= 300  ,RP(Artist)= 0.3421531452087008
K= 300  ,RP(Track)= 0.3525514403292177
K= 300  ,NDCG(Artist)= 0.19701615149768315
K= 300  ,NDCG(Track)= 0.017732168016096308
==========================================================
540 540 540
K= 400  ,RP(Artist)= 0.38602145796590226
K= 400  ,RP(Track)= 0.39663580246913543
K= 400  ,NDCG(Artist)= 0.20055373743954397
K= 400  ,NDCG(Track)= 0.0203292523610765
==========================================================
540 540 540
K= 500  ,RP(Artist)= 0.41816872427983515
K= 500  ,RP(Track)= 0.4306275720164606
K= 500  ,NDCG(Artist)= 0.20176398860520442
K= 500  ,NDCG(Track)= 0.02303619867021845
```

```
==========================================================
```

In [601]:
```
# Most similar playlist

def runMostSimilarPlaylistsModel(topKgrid):
    for topK in topKgrid:
        tracks_r_precision = []
        artists_r_precision = []
        ndcg_artists = []
        ndcg_tracks = []
        for row in testData:
            tracks_test = d['tracks']
            pid = d['pid']

            matches = [i[1] for i in mostSimilarUsers(pid, 25)]

#             pidMatch = match[0][1]
        #     print (match[0])
        #     print (pid, match[0][1])
            tracks = []
            for match in matches:
                if match in all_playlists.index:
                    tracks.extend(all_playlists.loc[match]['tracks'])
                elif match in all_playlists_test.index:
                    tracks.extend(all_playlists_test.loc[match]['track

            res = tracks[:topK]
#             print (res)
            artists_of_pred = getArtistsForTracks(res)
            tracks_r_precision.append(R_precision(res, row['withHeldTr
    #       print ("artists: ", artists_of_pred)
            artists_r_precision.append(R_precision(artists_of_pred, ro
            ndcg_artists.append(NDCG(row['withHeldArtists'], artists_o
            ndcg_tracks.append(NDCG(row['withHeldTracks'], res, topK))

        print (len(artists_r_precision), len(tracks_r_precision), len(
        print('K=',topK, ' ,RP(Artist)=',sum(artists_r_precision)/len(
        print('K=',topK, ' ,RP(Track)=',sum(tracks_r_precision)/len(tr
        print('K=',topK, ' ,NDCG(Track)=',sum(ndcg_tracks)/len(ndcg_tr
        print('K=',topK, ' ,NDCG(Artist)=',sum(ndcg_artists)/len(ndcg_

        # print('K=',topK, ' ,NDCG(Artist)=',NDCG(row['with_held_artis
        print('==========================================================')
```

In [602]:
```
artistsPerTrack['spotify:track:7wWw7hBmErNg9u5w6Xx0vy']
# all_tracks.loc['spotify:track:7wWw7hBmErNg9u5w6Xx0vy']
```

Out[602]:
```
{'spotify:artist:70ZTdbPEcEugBNay4MvxfL'}
```

In [603]:
```
runMostSimilarPlaylistsModel([10, 20, 30, 40, 50, 60, 70, 100, 200, 30
```

```
540 540 540
K= 10  ,RP(Artist)= 0.002628600823045268
K= 10  ,RP(Track)= 0.0012962962962962963
K= 10  ,NDCG(Track)= 0.003840966245713935
K= 10  ,NDCG(Artist)= 0.007977864938333077
==================================================
540 540 540
K= 20  ,RP(Artist)= 0.0062242798353909475
K= 20  ,RP(Track)= 0.0022222222222222222
K= 20  ,NDCG(Track)= 0.005922705302846955
K= 20  ,NDCG(Artist)= 0.014754076609815977
==================================================
540 540 540
K= 30  ,RP(Artist)= 0.008621399176954734
K= 30  ,RP(Track)= 0.003333333333333334
K= 30  ,NDCG(Track)= 0.007446768179943543
K= 30  ,NDCG(Artist)= 0.018070046324601904
==================================================
540 540 540
K= 40  ,RP(Artist)= 0.01314520870076425
K= 40  ,RP(Track)= 0.003703703703703704
K= 40  ,NDCG(Track)= 0.007857373687147416
K= 40  ,NDCG(Artist)= 0.025330212693239054
==================================================
540 540 540
K= 50  ,RP(Artist)= 0.013350970017636679
K= 50  ,RP(Track)= 0.003888888888888889
K= 50  ,NDCG(Track)= 0.007861812401158928
K= 50  ,NDCG(Artist)= 0.024026745646289267
==================================================
540 540 540
K= 60  ,RP(Artist)= 0.015763521457965893
K= 60  ,RP(Track)= 0.004444444444444445
K= 60  ,NDCG(Track)= 0.00843865362534843
K= 60  ,NDCG(Artist)= 0.026353638983939016
==================================================
540 540 540
K= 70  ,RP(Artist)= 0.015995002939447375
K= 70  ,RP(Track)= 0.004444444444444445
K= 70  ,NDCG(Track)= 0.008116940958575127
K= 70  ,NDCG(Artist)= 0.02579687035816271
==================================================
540 540 540
K= 100  ,RP(Artist)= 0.02325543797766019
K= 100  ,RP(Track)= 0.009999999999999998
K= 100  ,NDCG(Track)= 0.013494889433643386
K= 100  ,NDCG(Artist)= 0.029217733076365885
==================================================
540 540 540
```

```
K= 200  ,RP(Artist)= 0.05925631981187536

K= 200  ,RP(Track)= 0.01650205761316872
K= 200  ,NDCG(Track)= 0.017732648217199977
K= 200  ,NDCG(Artist)= 0.060037206960473405
==================================================
540 540 540
K= 300  ,RP(Artist)= 0.09401748971193429
K= 300  ,RP(Track)= 0.022242798353909448
K= 300  ,NDCG(Track)= 0.022526331691524182
K= 300  ,NDCG(Artist)= 0.08025832998776934
==================================================
540 540 540
K= 400  ,RP(Artist)= 0.1261816578483248
K= 400  ,RP(Track)= 0.03087448559670779
K= 400  ,NDCG(Track)= 0.030651736179761312
K= 400  ,NDCG(Artist)= 0.09407593063785434
==================================================
540 540 540
K= 500  ,RP(Artist)= 0.14376028806584398
K= 500  ,RP(Track)= 0.03698559670781893
K= 500  ,NDCG(Track)= 0.03398389929464602
K= 500  ,NDCG(Artist)= 0.09884490937419749
==================================================
```

In [499]:

```
def runPopularTrackModel(topKgrid):
  for topK in topKgrid:
    tracks_r_precision = []
    artists_r_precision = []
    ndcg_tracks = []
    for row in testData:
      res = getMostPopularSongs(topK, row['tracks'])
      artists_of_pred = getArtistsForTracks(res)
      tracks_r_precision.append(R_precision(res, row['withHeldTracks']
#       print ("artists: ", artists_of_pred)
      artists_r_precision.append(R_precision(artists_of_pred, row['wit
      ndcg_tracks.append(NDCG(row['withHeldTracks'], res, topK))

    print (len(artists_r_precision), len(tracks_r_precision), len(ndcg
    print('K=',topK, ' ,RP(Artist)=',sum(artists_r_precision)/len(arti
    print('K=',topK, ' ,RP(Track)=',sum(tracks_r_precision)/len(tracks
    print('K=',topK, ' ,NDCG(Track)=',sum(ndcg_tracks)/len(ndcg_tracks
    # print('K=',topK, ' ,NDCG(Artist)=',NDCG(row['with_held_artists']
    print('==================================================')
```

In [500]:
```
runPopularTrackModel([10, 15, 20, 25, 30,35, 50])
```

```
540 540 540
K= 10  ,RP(Artist)= 0.021813639035861233
K= 10  ,RP(Track)= 0.006502057613168728
K= 10  ,NDCG(Track)= 0.024607594573208557
==================================================
540 540 540
K= 15  ,RP(Artist)= 0.0367144326866549
K= 15  ,RP(Track)= 0.009650205761316873
K= 15  ,NDCG(Track)= 0.02926606691642524
==================================================
540 540 540
K= 20  ,RP(Artist)= 0.042660934744268075
K= 20  ,RP(Track)= 0.013441358024691348
K= 20  ,NDCG(Track)= 0.03677247019882888
==================================================
540 540 540
K= 25  ,RP(Artist)= 0.047610964138741906
K= 25  ,RP(Track)= 0.015174897119341554
K= 25  ,NDCG(Track)= 0.03814858523407897
==================================================
540 540 540
K= 30  ,RP(Artist)= 0.05747868900646676
K= 30  ,RP(Track)= 0.0170267489711934
K= 30  ,NDCG(Track)= 0.039792826801067764
==================================================
540 540 540
K= 35  ,RP(Artist)= 0.06000440917107582
K= 35  ,RP(Track)= 0.018693415637860058
K= 35  ,NDCG(Track)= 0.04180318043830286
==================================================
540 540 540
K= 50  ,RP(Artist)= 0.07232289829512056
K= 50  ,RP(Track)= 0.023878600823045232
K= 50  ,NDCG(Track)= 0.047050845121643835
==================================================
```

In [382]:
```
# Most similar Playlist

y_pred = []
y_pred_pid = []

for d in tqdm(testData):
    tracks_test = d['tracks']
    pid = d['pid']
    ms = []

    match = mostSimilarUsers(pid, 1)
```

```
    pidMatch = match[0][1]
#     print (match[0])
#     print (pid, match[0][1])
    tracks = []

    if pidMatch in all_playlists.index:
        tracks = all_playlists.loc[pidMatch]['tracks']
    elif pidMatch in all_playlists_test.index:
        tracks = all_playlists_test.loc[pidMatch]['tracks']

    y_pred_pid.append(pidMatch)
    for t in tracks_test:
        ms.extend(mostSimilar(t, 2, tracks))
#     for t in tracks:
# #         print (t)
#         ms.extend(mostSimilarUsers(t, 2))

    ms.sort(reverse=True)
#     print (ms)
#     print (len(ms))
    for i in range(len(ms)):
#         print (ms[i][1], tracks)
        if ms[i][1] not in tracks_test:
#             print ("W")
            y_pred.append(ms[i][1])
            break
    print (i, len(ms))
    if i == len(ms):
        print (ms)
```

```
 16%|██████                                          | 88/540 [00:00<00:04,
103.67it/s]

0 40
5 32
0 28
0 40
2 108
1 36
2 36
0 42
0 54
0 54
10 56
0 96
2 88
1 104
0 74
0 112
0 36
```

In [310]:
```
len(y_pred), len(testData), len(y_pred_pid)
```

Out[310]:
```
(521, 540, 540)
```

In [248]:
```
# Most similar tracks
y_pred = []
for d in tqdm(testData):
    tracks = d['tracks']
    ms = []
    artistHash = {}
    for t in tracks:
#         print (t)
        artistHash[]
        ms.extend(mostSimilarUsers(t, 2))

    ms.sort(reverse=True)
#     print (ms)
    for i in range(len(ms)):
        if ms[i][1] not in tracks:
            y_pred.append(ms[i][1])
            break
```

```
100%|██████████████████████████████████████████| 540/540 [07:17<00:00,
1.23it/s]
```

In [329]:
```
all_tracks_test.loc['spotify:track:7kv8WogLfELHaKukL9IjZM']['artist_ur
```

Out[329]:
```
'spotify:artist:3ApUX1o6oSz321MMECyIYd'
```

In [396]:
```
def acc_similarityWithLabel(pred, label):
    match = 0
    for i, j in zip(pred, label):
#         print (j)
        s = mostSimilar(j, 30)
        simTracks = [x for y, x in s]
#         print (simTracks)
        if i in simTracks:
            print (i, simTracks)
            match += 1

    return match / len(pred)
```

In [335]:
```
def acc_mostPopularArtist(pred, label):
    match = 0
    for i in label:
#         print ("A " + i + str(all_tracks_test.loc[i]['artist_uri']))
        if 'spotify:artist:3TVXtAsR1Inumwj472S9r4' == str(all_tracks_t
            match += 1

    return match / len(pred)
```

In [316]:
```
def acc_mostSimilarPlaylist(pred, label):
    match = 0
    for i, j in zip(pred, label):
        if i in playlistsPerTrack[j]:
            match += 1

    return match / len(pred)
```

In [249]:
```
def getAccuracy(predictions, labels):
    match = 0
    for i, j in zip(predictions, labels):
        if i in all_tracks.index:
            check = all_tracks.loc[i]['artist_uri']
        elif i in all_tracks_test.index:
            check = all_tracks_test.loc[i]['artist_uri']

        print (check, all_tracks_test.loc[j]['artist_uri'])

        if i == j or str(all_tracks_test.loc[j]['artist_uri']) == str(
            match += 1
#         if str(all_tracks_test.loc[j]['artist_uri']) == str(all_trac
#             match += 1
    return match / len(predictions)
```

In [374]:
```
all_playlists_test
```

Out[374]:

pid	name	collaborative	pid	modified_at	num_tracks	num_albums	num_followe
440000	mood	false	440000	1506384000	23	14	
440002	throwback thursday	false	440002	1459382400	32	25	
440007	party!!!	false	440007	1446249600	53	46	
440009	reggae	false	440009	1445904000	24	13	
440011	Sad Day	false	440011	1467504000	53	43	
...	...	...	...	...	...	...	
440989	slapz	false	440989	1508630400	28	22	
440990	romantic	false	440990	1505520000	57	53	
440991	my jams	false	440991	1506556800	52	47	
440993	Soundtracks	false	440993	1502323200	15	13	
440995	equilibrium	false	440995	1475884800	37	33	

540 rows × 12 columns

In [384]:
```
all_tracks_test.loc['spotify:track:7kv8WogLfELHaKukL9IjZM']
playlistsPerTrack['spotify:track:7kv8WogLfELHaKukL9IjZM']
for _, p in all_playlists_test.iterrows():
#     print (p)
    if 'spotify:track:7kv8WogLfELHaKukL9IjZM' in p['tracks']:
        print (p['pid'])
```

```
440000
```

In [397]:
```
acc_similarityWithLabel(y_pred, y_test)
```

```
spotify:track:0dA2Mk56wEzDgegdC6R17g ['spotify:track:7qiZfU4dY1lWllzX
7mPBI3', 'spotify:track:0dA2Mk56wEzDgegdC6R17g', 'spotify:track:6520a
j0B4FSKGVuKNsOCOi', 'spotify:track:6RUKPb4LETWmmr3iAEQktW', 'spotify:
track:3NdDpSvN911VPGivFlV5d0', 'spotify:track:4TZy1wLyHec06pwgFYDh1a'
, 'spotify:track:1NDxZ7cFAo481dtYWdrUnR', 'spotify:track:6D0b04NJIKfE
Mg040WioJQ', 'spotify:track:104buTcnP2AsxqB7U1FIZ4', 'spotify:track:0
4DwTuZ2VBdJCCC5TROn7L', 'spotify:track:72jbDTw1piOOj770jWNeaG', 'spot
ify:track:4nMlau89VAjmV7agkl7OY3', 'spotify:track:6kex4EBAj0WHXDKZMEJ
aaF', 'spotify:track:18DjMlzOhoG3hLgtn2kdhf', 'spotify:track:67citk3u
zWs5qbaIVKTeg8', 'spotify:track:3eR23VReFzcdmS7TYCrhCe', 'spotify:tra
ck:21TdkDRXuAB3k90ujRU1et', 'spotify:track:5hYTyyh2odQKphUbMqc5gN', '
spotify:track:5CtI0qwDJkDQGwXD1H1cLb', 'spotify:track:4uoumbAMEMaKdti
v763jKz', 'spotify:track:4iLqG9SeJSnt0cSPICSjxv', 'spotify:track:7675
gjlUZzneYiMrQ9Inx8', 'spotify:track:04CttTezSnv71USiiG9mIo', 'spotify
:track:4tCtwWceOPWzenK2HAIJSb', 'spotify:track:79cuOz3SPQTuFrp8WgftAu
', 'spotify:track:4b4KcovePX8Ke2cLIQTLM0', 'spotify:track:27SdWb2rFzO
6GWiYDBTD9j', 'spotify:track:78rIJddV4X0HkNAInEcYde', 'spotify:track:
3DXncPQOG4VBw3QHh3S817', 'spotify:track:1x5sYLZiu9r5E43kMlt9f8']
```

Out[397]:
```
0.0019193857965451055
```

In [386]:
```
playlistsPerTrack['spotify:track:7kv8WogLfELHaKukL9IjZM']
```

Out[386]:
```
{440000}
```

In [314]:
```
acc_mostSimilarPlaylist(y_pred_pid, y_test)
```

Out[314]:
```
0.03333333333333333
```

In [336]:
```
acc_mostPopularArtist(y_pred, y_test)
```

Out[336]:
```
0.009596928982725527
```

In [234]:
```
all_tracks_test = all_tracks_test.drop(columns='pos')
```

In [237]:
```
all_tracks_test.loc['spotify:track:0qOnSQQF0yzuPWsXrQ9paz']#['artist_u
```

Out[237]:
```
artist_name                                      Gym Class Heroes
track_uri                  spotify:track:0qOnSQQF0yzuPWsXrQ9paz
artist_uri                spotify:artist:4IJczjB0fJ04gs4uvP0Fli
track_name     Stereo Hearts (feat. Adam Levine) - feat. Adam...
album_uri                  spotify:album:2mumCpGmuE9iDeOvMx6XrB
duration_ms                                               210960
album_name                              The Papercut Chronicles II
Name: spotify:track:0qOnSQQF0yzuPWsXrQ9paz, dtype: object
```

In [286]:
```
getAccuracy(y_pred, y_test)
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
/var/folders/k1/cn4r859n1d3gq3qrhj73qzsh0000gn/T/ipykernel_2132/3912645101.py in <module>
----> 1 getAccuracy(y_pred, y_test)

/var/folders/k1/cn4r859n1d3gq3qrhj73qzsh0000gn/T/ipykernel_2132/1798252525.py in getAccuracy(predictions, labels)
     13 #         if str(all_tracks_test.loc[j]['artist_uri']) == str(all_tracks_test.loc[i]['artist_uri']):
     14 #             match += 1
---> 15     return match / len(predictions)

ZeroDivisionError: division by zero
```

In [151]:
```
sum([i == j for i, j in zip(y_pred, y_test)]) / len(y_pred)
```

Out[151]:
```
0.003703703703703704
```

In [236]:
```
all_tracks_test = all_tracks_test.drop_duplicates(subset='track_uri')
```

In [157]:
```
all_tracks.loc['spotify:track:5hTpBe8h35rJ67eAWHQsJx']
```

Out[157]:
```
pos                                                3
artist_name                                    Aminé
track_uri       spotify:track:5hTpBe8h35rJ67eAWHQsJx
artist_uri     spotify:artist:3Gm5F95VdRxW3mqCn8RPBJ
track_name                                  Caroline
album_uri       spotify:album:3lajefIuUk4SfzqVBSJy8p
duration_ms                                   209640
album_name                               Good For You
popularity                                         48
Name: spotify:track:5hTpBe8h35rJ67eAWHQsJx, dtype: object
```

In [153]:
```
[(i, j) for i, j in zip(y_pred, y_test)]
```

Out[153]:
```
[('spotify:track:5tz69p7tJuGPeMGwNTxYuV',
  'spotify:track:7kv8WogLfELHaKukL9IjZM'),
 ('spotify:track:0v9Wz8o0BT8DU38R4ddjeH',
  'spotify:track:0qOnSQQF0yzuPWsXrQ9paz'),
 ('spotify:track:5hTpBe8h35rJ67eAWHQsJx',
  'spotify:track:6pf9datdAtxQj4EU8UYHSx'),
 ('spotify:track:2d8JP84HNLKhmd6IYOoupQ',
  'spotify:track:6dViIgkYIRccNImPln58rS'),
 ('spotify:track:0QsvXIfqM0zZoerQfsI9lm',
  'spotify:track:7Clb8su2UxUJ4CQUtDW2UZ'),
 ('spotify:track:0v9Wz8o0BT8DU38R4ddjeH',
  'spotify:track:12sAo6GbNV3oha2raSxqGy'),
 ('spotify:track:0VgkVdmE4gld66l8iyGjgx',
  'spotify:track:5uDASfU19gDxSjW8cnCaBp'),
 ('spotify:track:1xznGGDReH1oQq0xzbwXa3',
  'spotify:track:3YBYCxerAPKmi7mxxspJ21'),
 ('spotify:track:152lZdxL1OR0ZMW6KquMif',
  'spotify:track:4Tid4MwqgR1CfKCun3tFon'),
 ('spotify:track:5CtI0qwDJkDQGwXD1H1cLb',
```

In [ ]:
```
pIds,trackIds = {},{}
reviewsPerUser = defaultdict(list)
filteredData = []
c = 0
for d in data:
    if userInteractionCounts[d['user_id']] < 3:
        continue
    u,i = d['user_id'],d['book_id']
    c += 1
    if not u in pIds: pIds[u] = len(pIds)
    if not i in trackIds: trackIds[i] = len(trackIds)
    # print (type(u))
    d['parsed_date_updated'] = dateutil.parser.parse(d['date_updated']
    reviewsPerUser[u].append(d)
    filteredData.append(d)

nUsers,nItems = len(userIDs),len(itetrackIdsmIDs)
```

In [ ]:
```
sortedReviewsPerUser = {x:sorted(reviewsPerUser[x], key=lambda x: x['p
```

In [337]:
```
e
```

Out[337]:

	pos	artist_name	track
spotify:artist:3TVXtAsR1Inumwj472S9r4	0	Drake	spotify:track:343YBumqHu19cGoGARU
spotify:artist:5K4W6rqBFWDnAN6FQUkS6x	16	Kanye West	spotify:track:2nBI3iWLhupR7LyAJ5G
spotify:artist:6eUKZXaKkcviH0Ku9w2n3V	45	Ed Sheeran	spotify:track:2RttW7RAu5nOAfq6YFv
spotify:artist:2YZyLoL8N0Wb9xBt1NhZWg	36	Kendrick Lamar	spotify:track:7KXjTSCq5nL1LoYtL7X
spotify:artist:5pKCCKE2ajJHZ9KAiaK11H	27	Rihanna	spotify:track:42Ow7PS3YtCWplolUUi
...	...	...	
spotify:artist:2bl0Qv6eSrkSw82UCFQCCS	3	Mega Banton	spotify:track:3sbmfsbvJdAEIEYbISNU
spotify:artist:6huJo1XyJ0v3d5WJ5G1cYc	11	Steam	spotify:track:0gyC5TwpZmZWNKELnpl
spotify:artist:0kJMPTXq7h3ztpDukSx5iD	3	Pras	spotify:track:31bf9SEOppLU6lQ85d8
spotify:artist:6ENAap6kOirsqATazyWBEP	0	David Dallas	spotify:track:09WvikcE8OuI7Si8Gj
spotify:artist:2OaHYHb2XcFPvqL3VsyPzU	49	Rico Nasty	spotify:track:2hcO4itSiCuWdsNWdVB

12336 rows × 9 columns

In [338]:
```
mp_tracks
```

Out[338]:

track_uri	pos	artist_name	tra
spotify:track:7BKLCZ1jbUBVqRi2FVlTVw	24	The Chainsmokers	spotify:track:7BKLCZ1jbUBVqRi2F
spotify:track:7KXjTSCq5nL1LoYtL7XAwS	36	Kendrick Lamar	spotify:track:7KXjTSCq5nL1LoYtL
spotify:track:152lZdxL1OR0ZMW6KquMif	28	Khalid	spotify:track:152lZdxL1OR0ZMW6l
spotify:track:3DXncPQOG4VBw3QHh3S817	1	DJ Khaled	spotify:track:3DXncPQOG4VBw3QHh
spotify:track:0SGkqnVQo9KPytSri1H6cF	21	Big Sean	spotify:track:0SGkqnVQo9KPytSri
...	...	...	
spotify:track:4ly644j0PpHssDJ33tXbIS	46	David Bisbal	spotify:track:4ly644j0PpHssDJ3
spotify:track:4p2AuPOS7ptkDgDI7Unmzg	45	Maite Perroni	spotify:track:4p2AuPOS7ptkDgDI7
spotify:track:2VRssqds7rRNb4eaG9FaY8	44	Jencarlos	spotify:track:2VRssqds7rRNb4eaG
spotify:track:03h8eaVgdJJ14cCKiO4ONX	43	Joey Montana	spotify:track:03h8eaVgdJJ14cCKiO
spotify:track:7jgBTiVeF54ccOHGTBkibB	50	Gucci Mane	spotify:track:7jgBTiVeF54ccOHGT

40399 rows × 9 columns

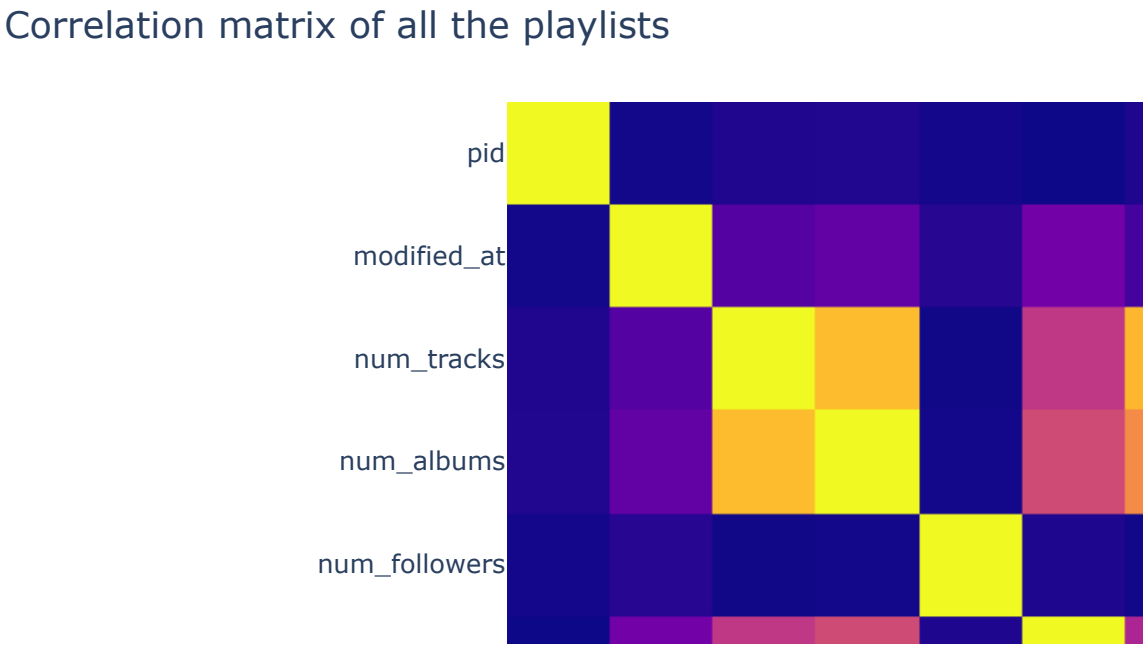
In [339]:
```
mostSimilar('spotify:track:7BKLCZ1jbUBVqRi2FVlTVw', 10)
```

Out[339]:
```
[(0.25961538461538464, 'spotify:track:6DNtNfH8hXkqOX1sjqmI7p'),
 (0.22448979591836735, 'spotify:track:4pdPtRcBmOSQDlJ3Fk945m'),
 (0.17647058823529413, 'spotify:track:6875MeXyCW0wLyT72Eetmo'),
 (0.1568627450980392, 'spotify:track:4pLwZjInHj3SimIyN9SnOz'),
 (0.15, 'spotify:track:4Hf7WnR761jpxPr5D46Bcd'),
 (0.15, 'spotify:track:2XMTqoHHSH0lvuXrvIEdco'),
 (0.14035087719298245, 'spotify:track:6WhzFzROw3aq3rPWjgYlxr'),
 (0.13978494623655913, 'spotify:track:0prNGof3XqfTvNDxHonvdK'),
 (0.128, 'spotify:track:0QsvXIfqM0zZoerQfsI9lm'),
 (0.12280701754385964, 'spotify:track:0azC730Exh71aQlOt9Zj3y')]
```

In [341]:
```
all_tracks.loc['spotify:track:4pdPtRcBmOSQDlJ3Fk945m']
```

Out[341]:
```
pos                                                    12
artist_name                                     DJ Snake
track_uri       spotify:track:4pdPtRcBmOSQDlJ3Fk945m
artist_uri     spotify:artist:540vIaP2JwjQb9dm3aArA4
track_name                                Let Me Love You
album_uri       spotify:album:55bbXORm6ZrVq52zfZnxBf
duration_ms                                       205946
album_name                                        Encore
popularity                                            32
Name: spotify:track:4pdPtRcBmOSQDlJ3Fk945m, dtype: object
```

In [402]:
```
fig = px.imshow(all_playlists.corr())
fig.update_layout(
        title="Correlation matrix of all the playlists")
fig.show()
```



In [ ]:

In [400]:
```
all_tracks
```

Out[400]:

	track_uri	artist_uri	track_name	
	ımCchFjTJoVwymcRmiLC	spotify:artist:0gXx2aQ2mfovDfqCw10MQC	Thunderdome	spotify:album:4
	ePQ29ix8gC0CbRHcGoBz	spotify:artist:0gXx2aQ2mfovDfqCw10MQC	Zig Zak	spotify:album:7
	q8RT5Kd3ExOGVTiUQbR	spotify:artist:4iVhFmG8YCCEHANGeUUS9q	Hot Like Sauce	spotify:album:0li
	5LFzH1i152LuEUxNZJAD	spotify:artist:4iVhFmG8YCCEHANGeUUS9q	One Day They'll Know	spotify:album:4l
	l3MOnnIhaT6MsRqNu13q	spotify:artist:4iVhFmG8YCCEHANGeUUS9q	Done Wrong (Opiuo Remix)	spotify:album:0pl
	...	...	...	
	37Aek2DGI5s4WNAh8zKR	spotify:artist:2EMAnMvWE2eb56ToJVfCWs	502 Come Up	spotify:album:6
	'dJOlGtU6ABq7tJNDhh8V	spotify:artist:4wf83MgUJNKr3Th3MYBEOz	How You Gonna Act Like That	spotify:album:2
	zP2KMwrhAdZEtIyp8Z8w	spotify:artist:46GXASE9LHzyssNqKOInUu	Jumpshot	spotify:album
	O4itSiCuWdsNWdVBKD9	spotify:artist:2OaHYHb2XcFPvqL3VsyPzU	Hey Arnold (Remix)	spotify:album:2
	jgBTiVeF54ccOHGTBkibB	spotify:artist:13y7CgLHjMVRMDqxdx0Xdo	Stutter	spotify:album:4MoV