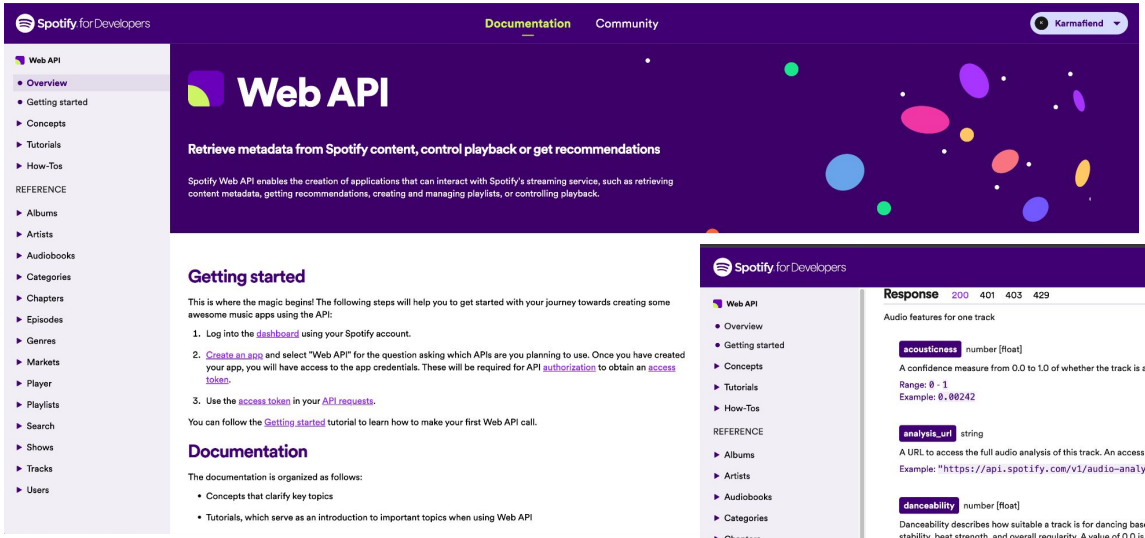
Data Bootcamp

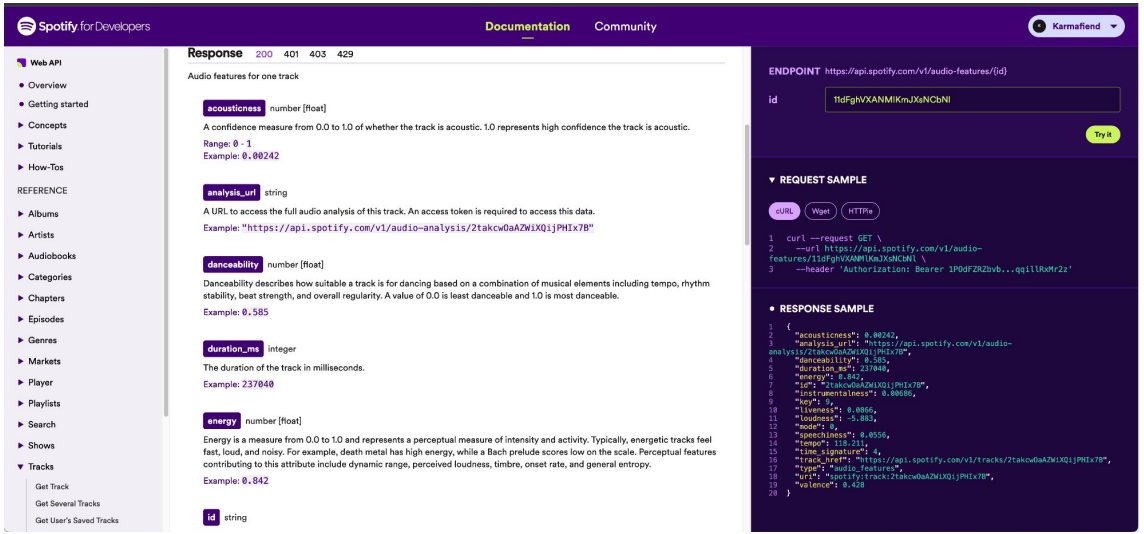
Project #1 - Group #10 presentation
3/25/24

Group #10: Team Member Introductions

- Chris Kilkes
- Divya Govil
- Ryan Hoffman

Step 1: Work with the Spotify API





For more information visit the project README file located here:
https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Project & Analysis overview

Analysis Goal

Examine musical attributes globally

- Analyze detailed differences about Spotify user artist & preference (popularity) by country, artist, genre and other criteria

Analysis Result: Three types of questions

Those we could analyze

Music attributes like “danceability” and “energy”

Those we could <u>not</u> analyze

Song, artist popularity by country, genre, etc. (i.e. our original goal)

Those to analyze further

Global and country attribute min/max, individual artist comparison

Technical Logistics & Challenges Addressed

- **Logistics:**
  - APIs used: Spotify's "Web API" (https://developer.spotify.com/documentation/web-api)
  - Tools used: Pearson R from Scipy.stats library (not taught, but used in Challenge 6)
- **Technical challenges:**
  - API often did not provide the data in the format, or in its content, to support our analyses (i.e. forced to focus on aggregated music attributes, rather than their popularity by country and artist as we desired.
  - Spotify token process to access API (not taught)
  - Installing Spotipy library in order to access Spotify API (not taught)
  - Modifying Mac/Win shell to store Spotify credentials to anonymize code (not taught)

For more information visit the project README file located here:
https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Step 2: Secure a New Source Of Data - Top 50 Playlists

1.

```
#Display Top 50 playlist by country
def search_playlist(result, query): # returns the playlist id from the result of the search method
    try:
        # Make sure the name of the playlist is the same one that we searched for and that it is made by Spotify (and not
        # a user). Make both sides lowercase to make the comparison case-insenstive.
        if(str.lower(result['playlists']['items'][0]['name']) == str.lower(query) and
           result['playlists']['items'][0]['owner']['id'] == 'spotify'):
            playlist_id = result['playlists']['items'][0]['id']
            return playlist_id
        else:
            print('No playlist found for: ' + query)
    # While some countries do not return a playlist, it appears one or two return a playlist but that
    # has no data in it, so we have to catch the IndexError this situation causes.
    except IndexError as e:
        print(e)

# Had to hard code the list of countries that was outputted as a result of the prior cell because a lot of countries
# had extraneous words in them e.g. "republic of ..."
countries = ['Andorra',
 'United Arab Emirates',
 'Antigua and Barbuda',
 'Albania',
 'Armenia',
 'Angola',
 'Argentina'
```

Leveraged existing Spotify code from Github repository to identify the Top 50 Playlists by country in Spotify - required hardcoding countries

Step 3: Pull data, Assign to Countries

2.

```
# Pull the 'Top 50 - <country name>' playlists for the countries in the list then create a dataframe and corresponding
# CSV for the playlist for each country.
for country in countries:
    # Search Spotify for the playlist (e.g. Top 50 - United States)
    search_result = sp.search(playlist_name + country, type='playlist', limit=1)
    # Pass the search results to the search_playlist function to get back a playlist ID which is required to
    # get the list of tracks in the playlist.
    playlist_id = search_playlist(search_result, playlist_name + country)
    # Only continue if the playlist ID search returned data.
    if playlist_id is not None:
        # Create list of tracks from the playlist to save into a CSV.
        playlist_tracks = sp.playlist_tracks(playlist_id, additional_types=('track'))
        track_ids = []
        track_names = []
        for track_obj in playlist_tracks['items']:
            # Get the track ids from the playlist and put it in a list
            # Use the ids to search songs and their features/analysis and then put that data in csv files by country
            track_ids.append(track_obj['track']['id'])
            track_names.append(track_obj['track']['name'])
        # export dataframe to csv before the first for loop moves to next country
        track_information_df = pd.DataFrame(sp.audio_features(track_ids))
        track_information_df.insert(0, 'name', track_names, False)
        track_information_df.to_csv('resources/' + country + '.csv', sep=',', index=False, encoding='utf-8')
        print(f'Finished converting {country}\'s Top 50 Playlist to csv')
```

Created list of tracks from the playlists to save into a .CSV file format and then we put that data into separate .csv files by country - this gave us country specific data

Step 4: Combine Country .CSVs into 1 combined dataframe

3.

```
[ ]: # Combine CSVs and calculate averages for audio features by country.

[4]: #Merge country data files into one combined dataframe "df_combined" for analysis
     # Use the Argentina.csv playlist as a sample to create a list of columns that are numerical (floats or ints) since
     # statistical anaylsis can only be done on numerical data. This list of columns is then used to both calculate
     # averages and then to be listed as the keys of the master dataframe that will combine the averages of every
     # column for every country.
     dict_combined = {"Country": []}
     list_columns = []
     df_current = pd.read_csv('resources/Argentina.csv')
     for column, dtype in df_current.dtypes.items():
         if dtype in ['float64', 'int64']:
             # Add the word "average" to the beginning of each column.
             dict_combined[f'average_{column}'] = []
             list_columns.append(column)

     # Iterate through every CSV created in the prior cell to calcualte the average of each column (i.e. audio feature) for
     # each CSV then add it to a master dictionary of all the averages that will later be converted into a dataframe.
     for file in os.listdir('resources'): # Lists every file in the directory.
         # Remove file extension to use the name as the country for the dataframe at the end.
         country = file.removesuffix('.csv')
         dict_combined["Country"].append(country)
         # Concatenate the name of the file with the name of the folder it is inside of (e.g. 'resources/Argentina.csv').
         df_current = pd.read_csv('resources/'+file)
         for column in list_columns:
             curr_avg = df_current[column].mean()
             dict_combined[f'average_{column}'].append(curr_avg)

     # Convert the dictionary created above into a dataframe.
     df_combined = pd.DataFrame.from_dict(dict_combined)

     # Set the country as the index to make it easier to later find the minumum and maximum values/countries.
     df_combined.set_index('Country', inplace=True)
     print("Global_Mean:\n", df_combined.mean(),"\n\n", sep='')
```

Use combined dataframe to then analyze country specific music attributes to determine statistical relationships (correlation, regression) and determine top / lowest minimum and maximum for specific countries

Spotify Web API Key Music Attribute Definitions

Acousticness	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
Danceability	Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
Energy	Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale.
Instrumentalness	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.
Loudness	The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Values typically range between -60 and 0 db.
Speechiness	Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value.
Valence	A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).

- **For the complete Web API music attribute definitions visit:**
  https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Global analysis

Global Analysis: Overview

In this first analysis, after discarding our original interest in analyzing user music prefernce by countries and regions, and instead focused on the global music attributes (“danceability,” “valence,” “energy,” etc.) as a whole.

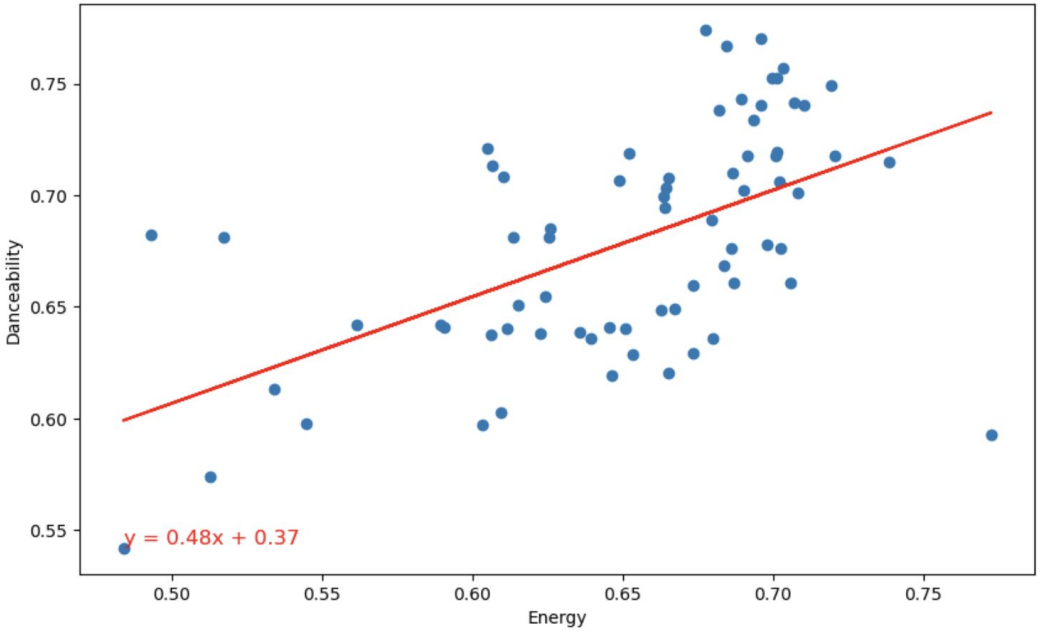
We conducted correlation and regression analyses on a series of music attribute comparison to determine if there were any statistical relationships:

- Most statistically significant: “Energy” relatively strongly predicts “Danceability”
- Somewhat less strongly, “Loudness” also predicts “Danceability”

Two other comparisons we made - “Loudness” vs. “Acousticness” they were only slightly negatively correlated; and “Speechiness” and “Valence” were only mildly correlated.

Global Analysis: Energy vs. Danceability

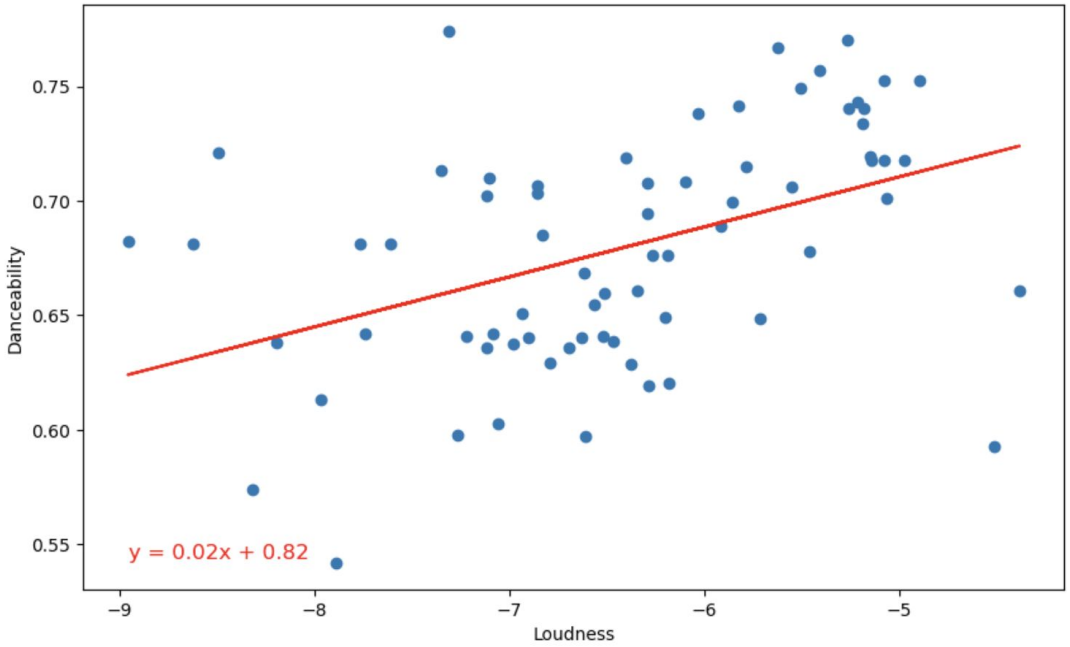
- **Finding**: As energy increases, danceability does too
  - Correlation: 0.5447
  - P-value: 0.0000
  - Regression: y = 0.48x + 0.37
- Globally there is a relatively strong, statistically significant relationship, indicating the more energy a song has the more danceable it tends to be too



- For the complete Web API music attribute definitions visit: https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Global Analysis: Loudness vs. Danceability

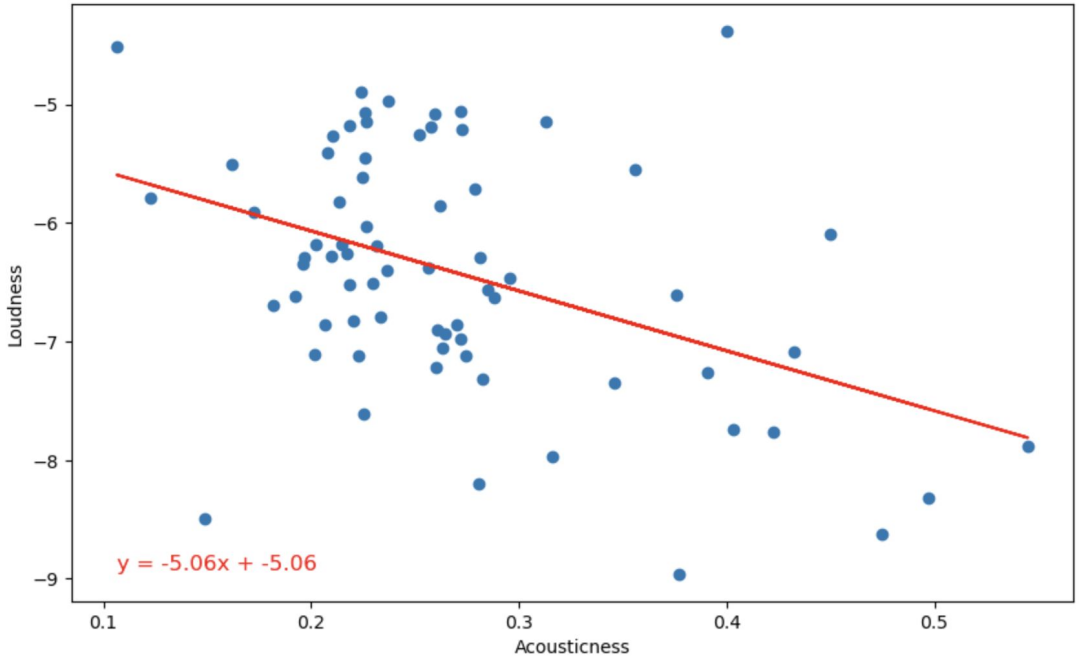
- **Finding**: Louder music may be slightly more danceable
  - Correlation: 0.4388
  - P-value: 0.0001
  - Regression: y = 0.02x + 0.82
- The relationship is relatively weak, though statistically significant, indicating that louder music may generally be more danceable



- For the complete Web API music attribute definitions visit: https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Global Analysis: Loudness vs. Acousticness

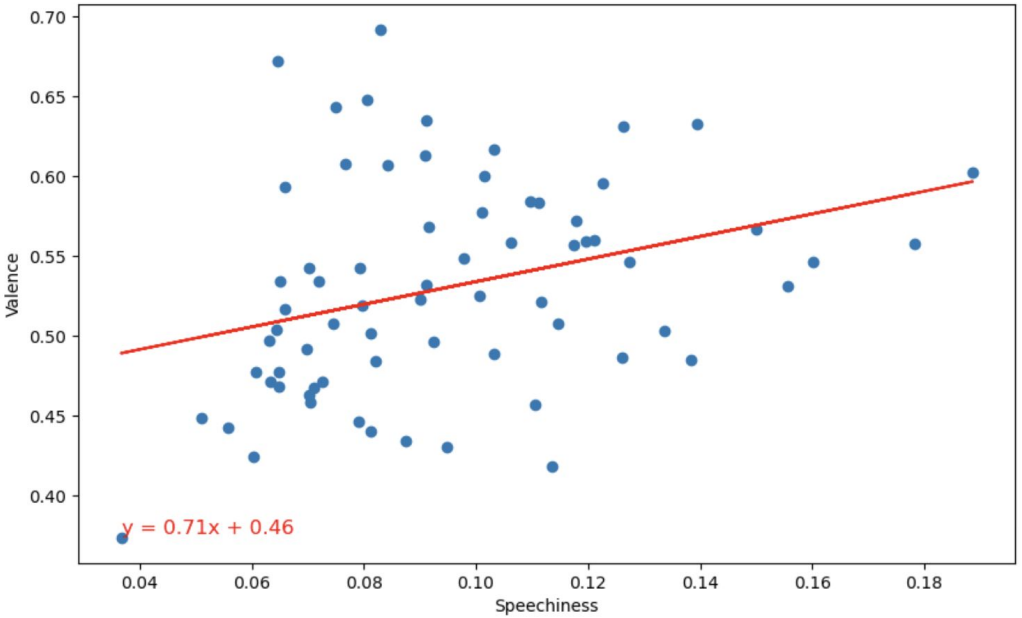
- **Finding**: The louder music is the slightly less likely it is to be acoustic
  - Correlation: -0.4229
  - P-value: 0.0003
  - Regression: y = -5.06 + -5.06
- While not a strong negative correlation, the analysis indicates that globally acoustic music may tend to be quieter than other forms of music (i.e. “heavy metal”)



- For the complete Web API music attribute definitions visit: https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Global Analysis: Speechiness vs. Valence

- **Finding**: The relationship between spoken word vs. positive “feeling” is moderate but statistically significant
  - Correlation: 0.3211
  - P-value: 0.0067
  - Regression: y = 0.71x + 0.46
- In theory this means that songs with more spoken word content may tend to bemore positive (i.e. “valence”)



- For the complete Web API music attribute definitions visit: https://developer.spotify.com/documentation/web-api/reference/get-audio-features

Regional analysis

Regional Analysis

In this section we grouped the available countries by geographic and cultural aggregate data and analyzed it on the features mentioned above

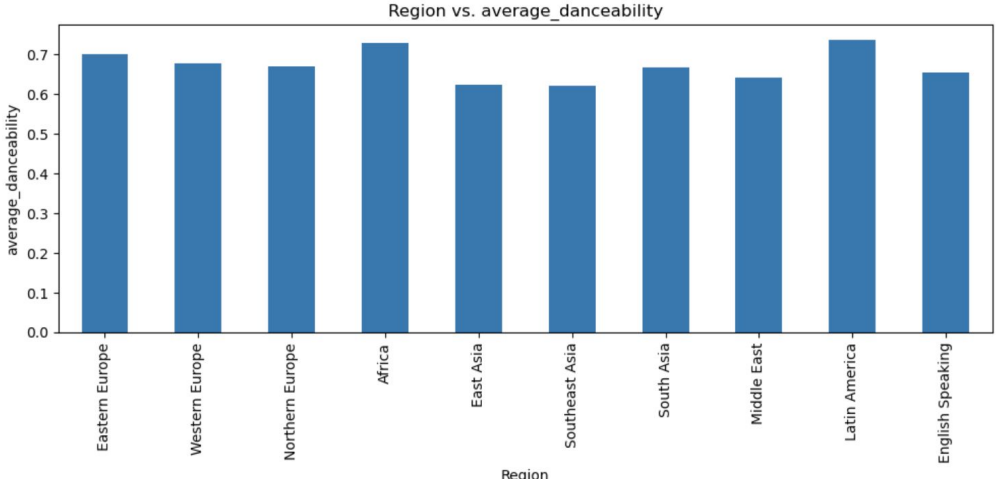
We were hoping to learn whether different regions in the world had preferences by way of key features like: danceability, loudness, speechiness, valence, and duration

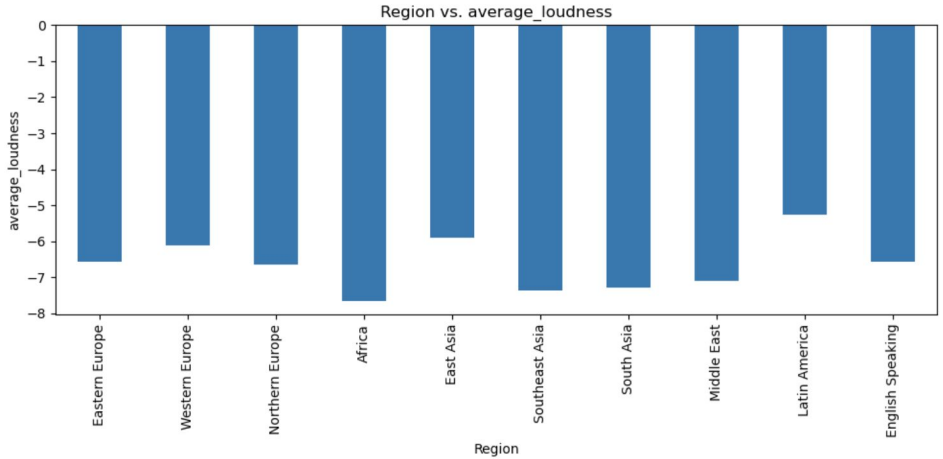
Liberties taken while grouping:

- Spotify lacks many countries top 50 playlists, inc: China, Russia, Jamaica, most African countries
- Some regions were grouped with other sparse regions: ‘English Speaking’ and South Asia
- Important to note that the playlists updated daily and are subject to decision by spotify’s algorithms

Regional Analysis

- Africa and Latin America
- Western, Northern, and Eastern Europe
- South, Southeast, and East Asia
- Middle East and Central Asia

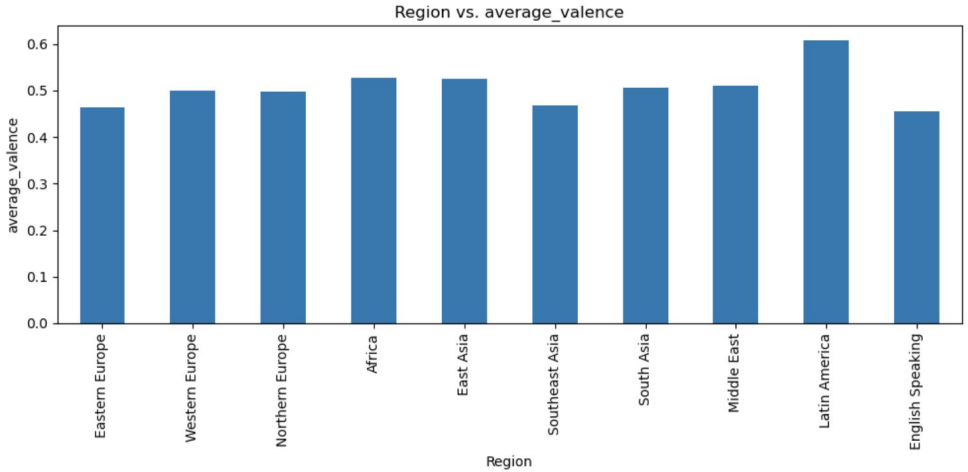




Regional Analysis

English Speaking Countries

- Below average danceability
- Above average loudness
- Below average speechiness
- Lowest average valence
- Above average in duration
- Below average instrumentalness



If these attributes are any of the reasons why you like english music, perhaps there is other music out there that better embodies why you enjoy listening, go explore!

Analysis questions requiring further investigation (and time)

Area of Exploration: Artist Comparison to Global Means

Taylor Swift
“Cruel Summer”

“Cruel Summer’s” Spotify music attributes:

- Danceability: 0.552
- Energy: 0.702
- Key: 9
- Loudness: 5.707
- Mode: 1
- Speechiness: 0.157
- Acousticness: 0.117
- Instrumentalness: 0.000021
- Liveness: 0.105
- Valence: 0.564
- Tempo: 169.994
- Duration_ms: 178427
- Time_signature: 4

- **Comparison of “Cruel Summer” to global means:**
  - Energy = 0.702, below max country average of 0.772
  - Key = 9, higher than max country average value of 6.86
  - Mode = 1, higher than max country average value of 0.9
  - Tempo = 169, higher than max country average value of 138.41
  - Time signature = 4, below max country average value of 4.08
  - Average liveness = 0.105, below the country average min of 0.125
- **Analysis Implication:**
  - Future statistical analysis suggested on whether specific music attributes influence a song’s popularity.

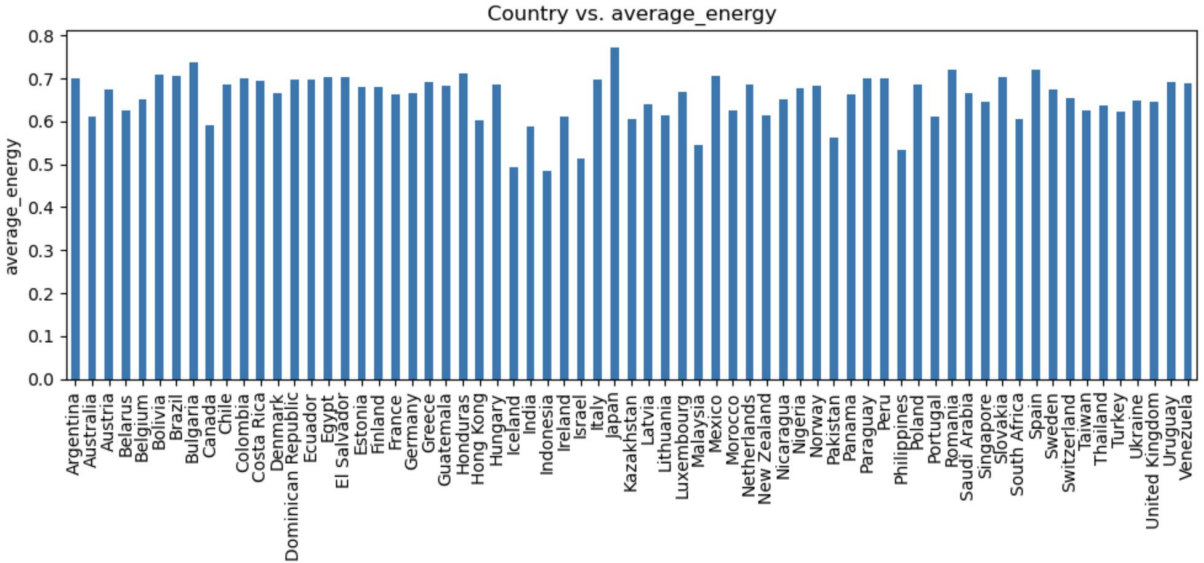
Area of Exploration: Global Min and Max comparisons

- **Potential analysis:**
  - Statistical comparison of individual country min / max
- **Analysis purpose:**
  - With additional contextual data (popularity, user demographics / psychographics) could use to determine if music and songs should be tailored to country music attribute preference

	Category	Max Country	Max Number	Min Country	Min Number
0	average_danceability	Nigeria	0.774220	Indonesia	0.542100
1	average_energy	Japan	0.772560	Indonesia	0.484000
2	average_key	Paraguay	6.860000	Pakistan	4.300000
3	average_loudness	Brazil	-4.378940	Iceland	-8.959100
4	average_mode	Indonesia	0.900000	Morocco	0.200000
5	average_speechiness	Dominican Republic	0.188632	Indonesia	0.036688
6	average_acousticness	Indonesia	0.544730	Japan	0.106504
7	average_instrumentalness	Saudi Arabia	0.139075	Italy	0.000113
8	average_liveness	Brazil	0.338552	South Africa	0.125004
9	average_valence	Uruguay	0.692160	Indonesia	0.373660
10	average_tempo	Bulgaria	138.416180	India	106.487200
11	average_duration_ms	South Africa	300000.140000	Ukraine	147690.400000
12	average_time_signature	Nigeria	4.080000	Mexico	3.460000

Area of Exploration: Country variance comparison

- **Potential analysis:**
  - Statistical comparison of individual country music attributes
- **Analysis purpose:**
  - Could use to determine if music and songs should be tailored to country specific music attributes



Thank you!

Appendix

Regional analysis

Regional Analysis

Africa and **Latin America** had the most dancy music on average but...

- Africa has the quietest and Latin America has the loudest average music
- Both are relatively speechy, Africa is the most
- Both are relatively high in valence (happy, euphoric), Latin America is the most
- Africa has the longest songs of any region on average, Latin America is amongst the shortest average song regions
- Africa was among the most instrumental, Latin America was the least

Regional Analysis

East Asia, **Southeast Asia**, and **South Asia**

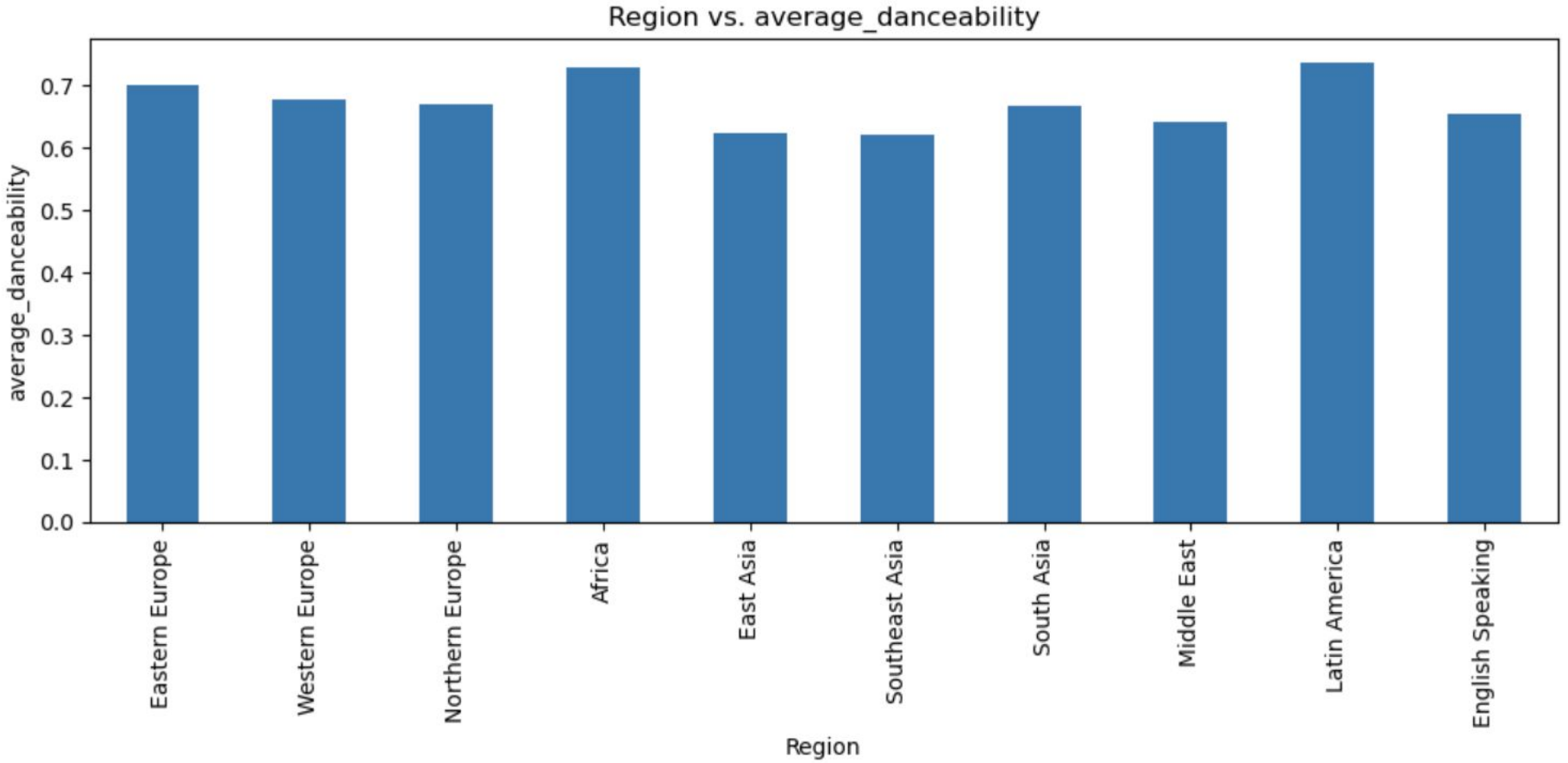
- East, Southeast, and South Asia were among the least dancy regions, South Asia is about at world average danciness
- East Asia is the second quietest, Southeast and South Asia are louder than average
- East and Southeast Asia are the least speechy and South Asia is more speechy than average
- All three had an average duration that was longer than the world average
- East and Southeast Asia had the 2nd and 3rd least instrumental songs on average, while South Asia had the most instrumental songs

Regional Analysis

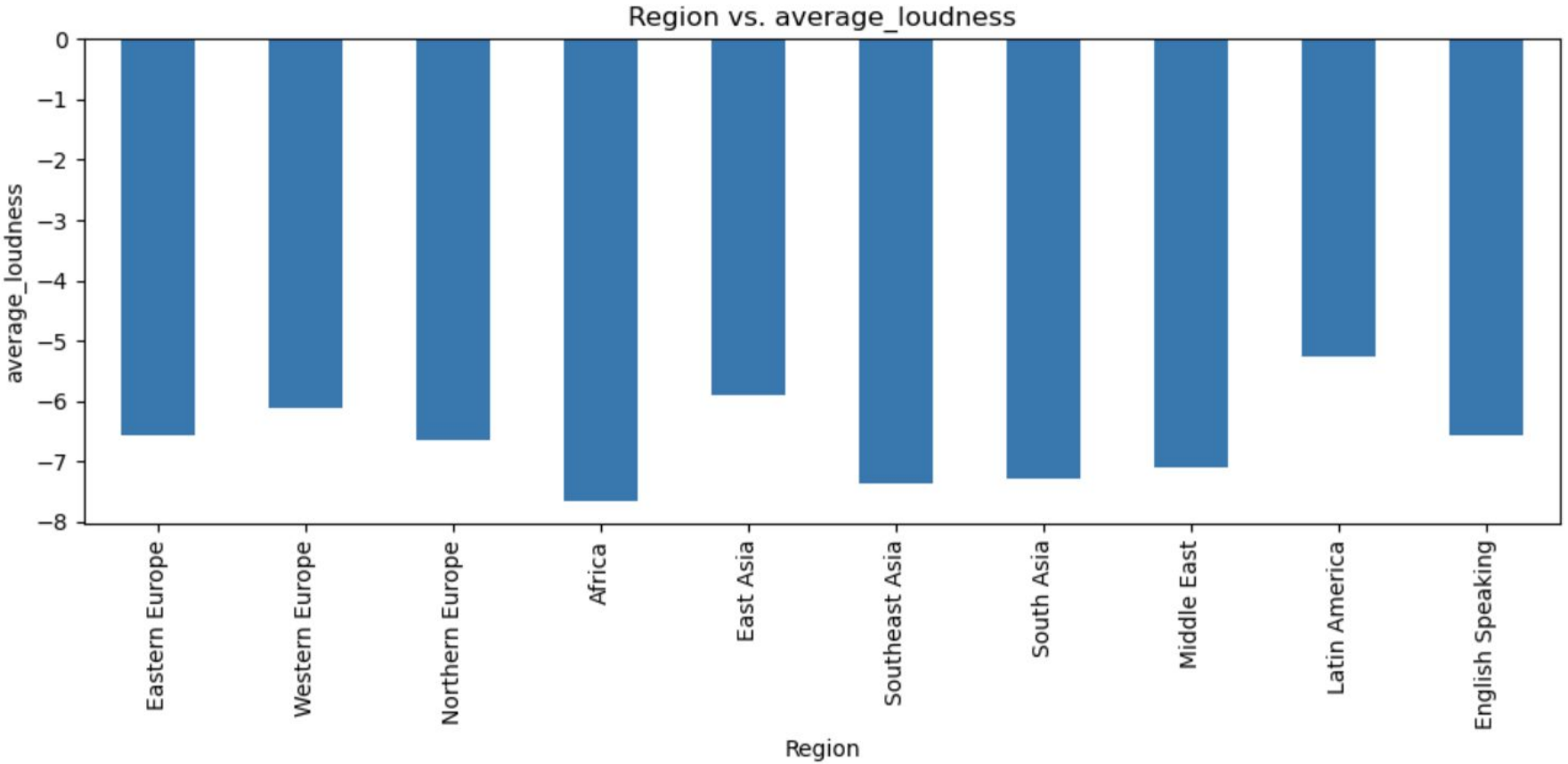
Eastern Europe, **Western Europe**, and **Northern Europe**

- All three were at or above world average danciness
- All three were at or above world average speechiness, but Eastern Europe was the 2nd most speechy
- All three were at or below world average valence, but Eastern Europe was the 2nd least valent(?) region in the world
- Together, these regions comprised the three shortest average duration regions
- Western and Northern Europe were average instrumentally, but Eastern Europe was one of the most instrumental (3rd)

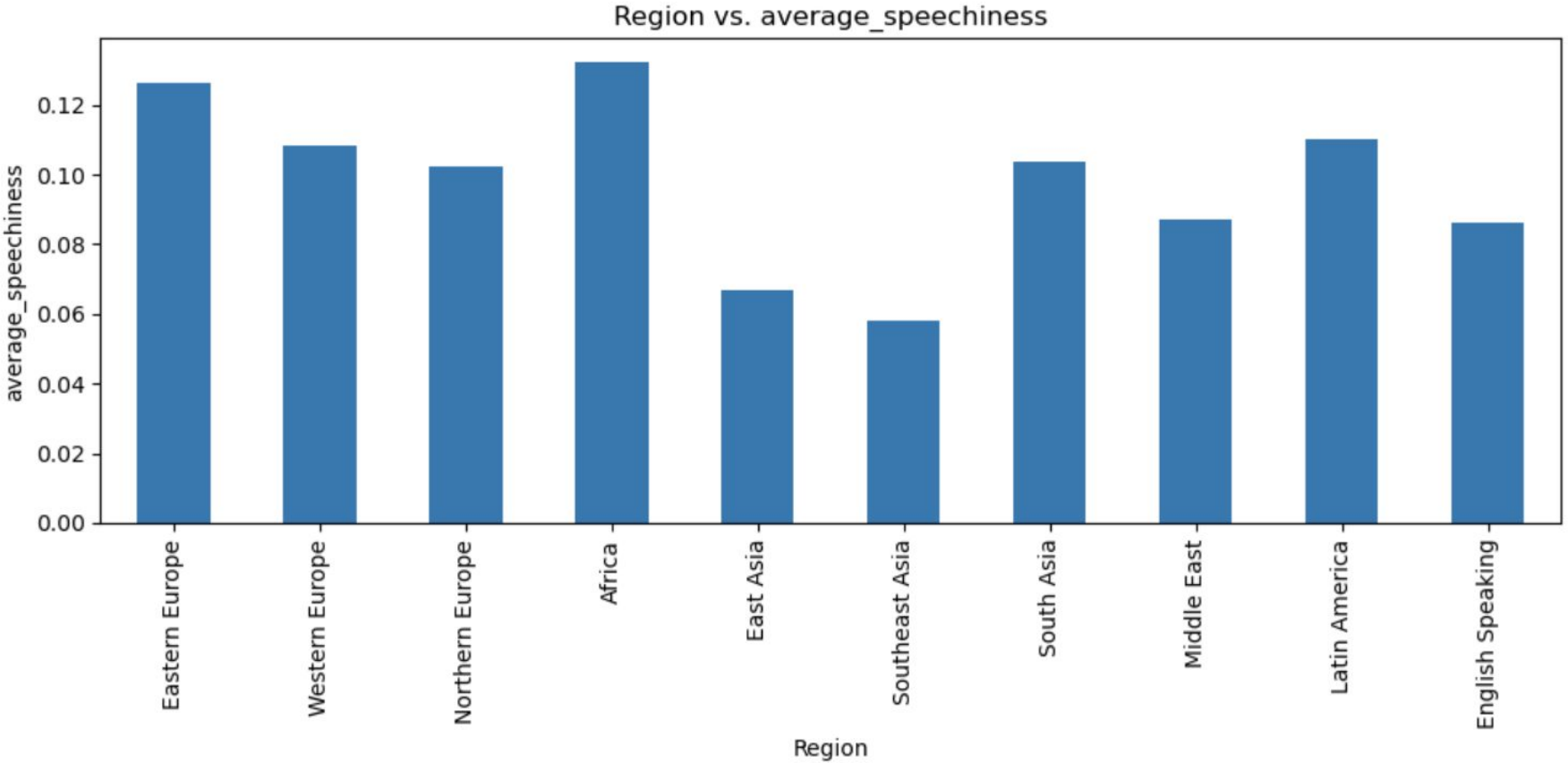
Regional Analysis



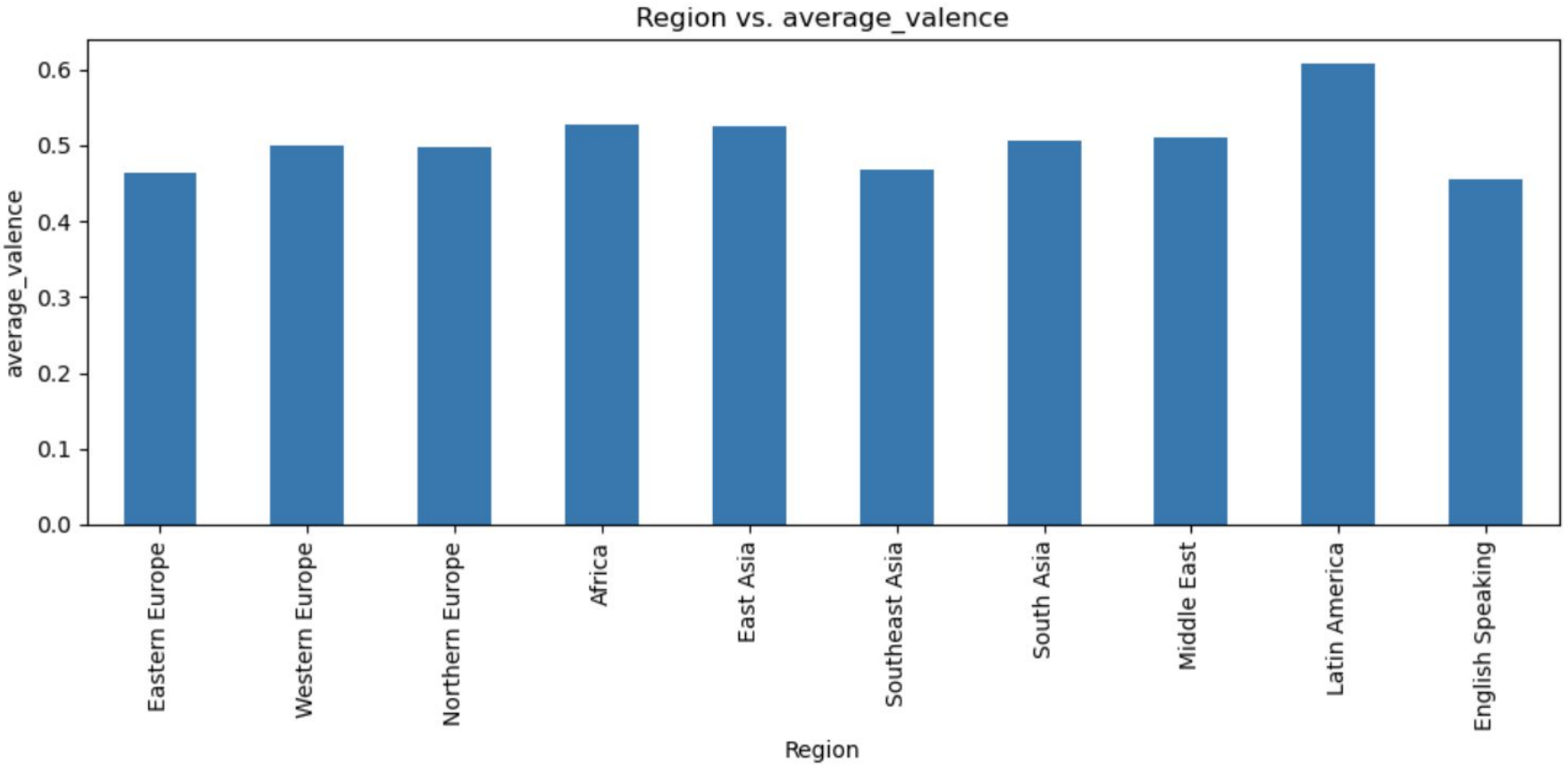
Regional Analysis



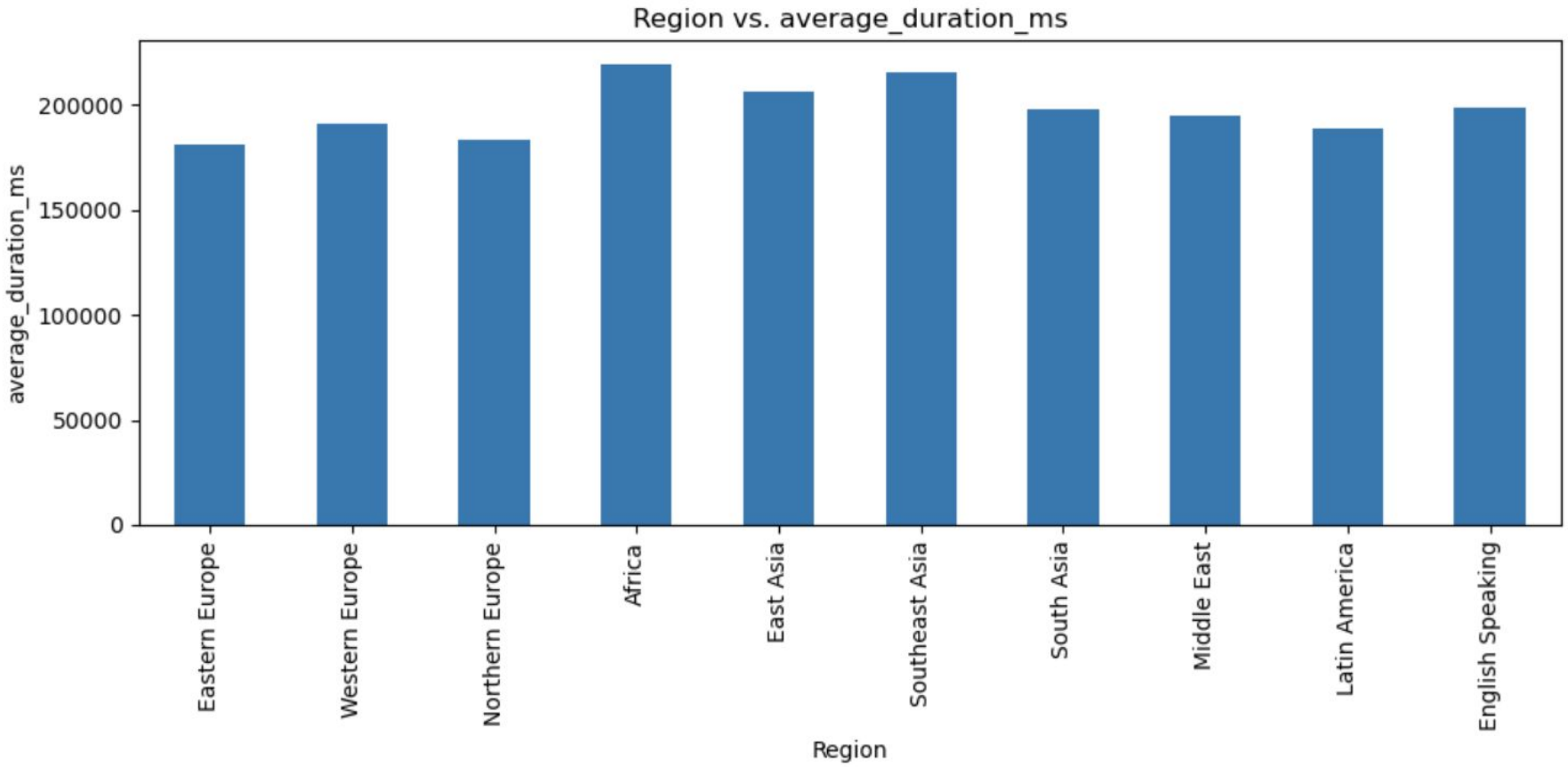
Regional Analysis



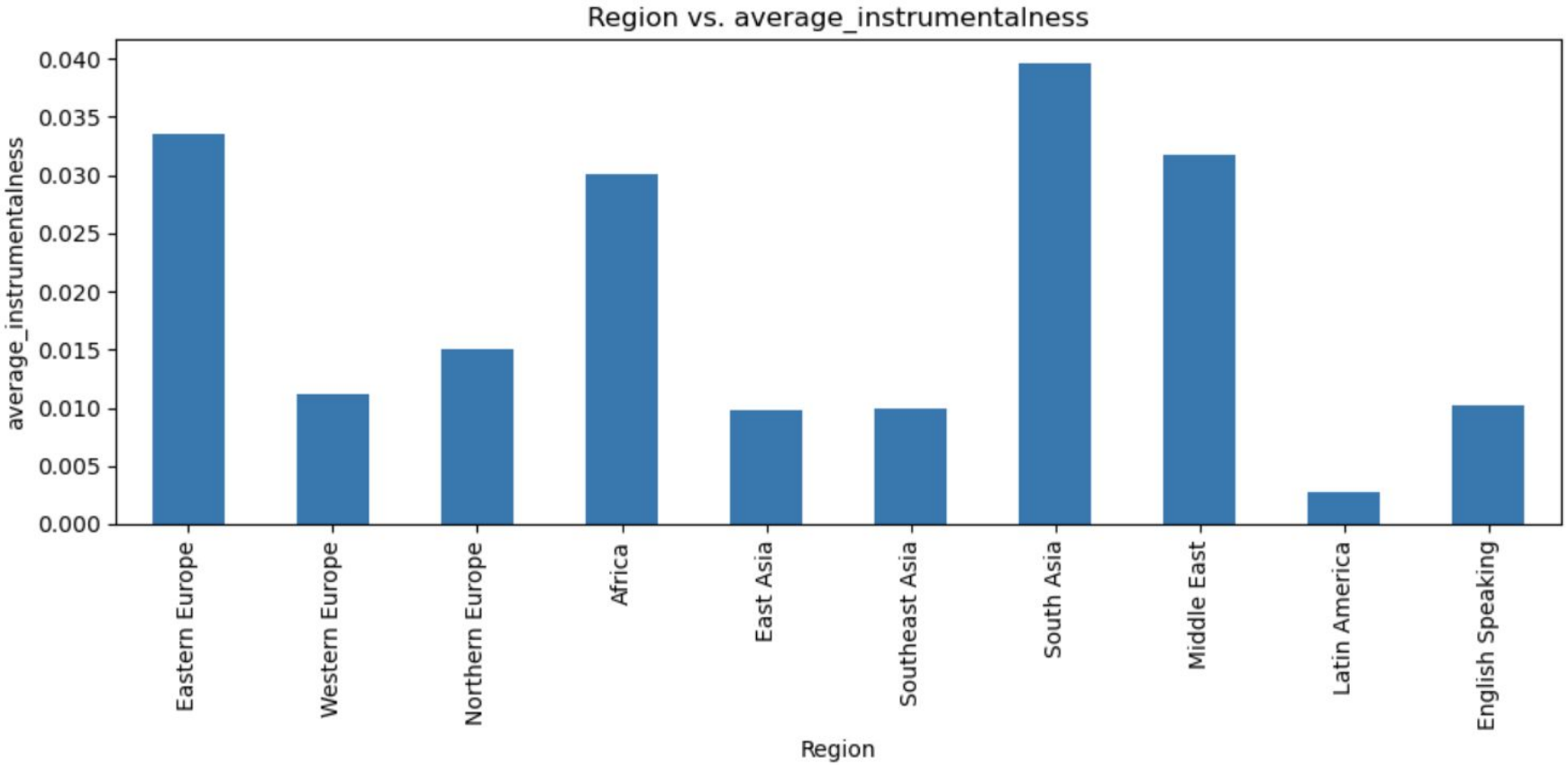
Regional Analysis



Regional Analysis

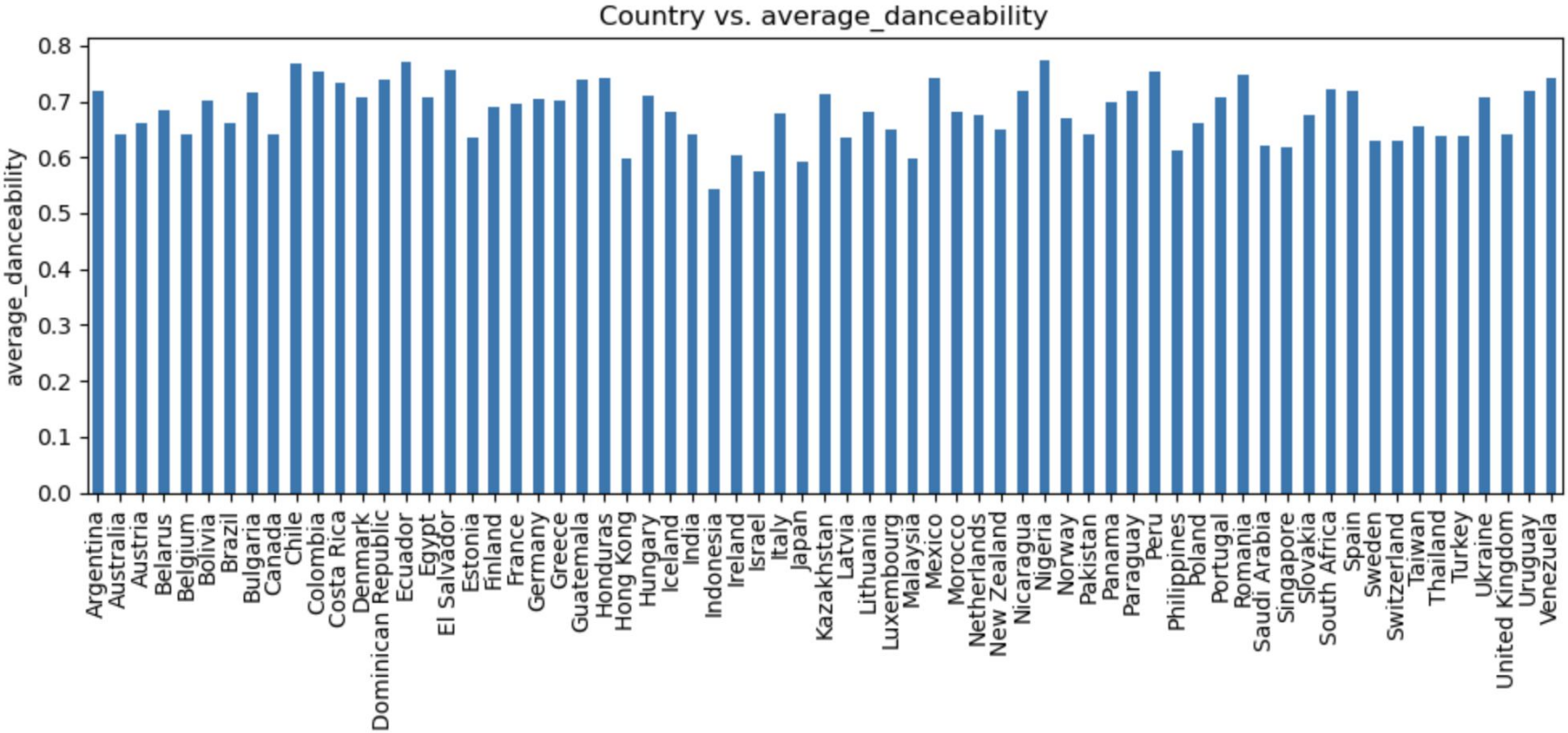


Regional Analysis

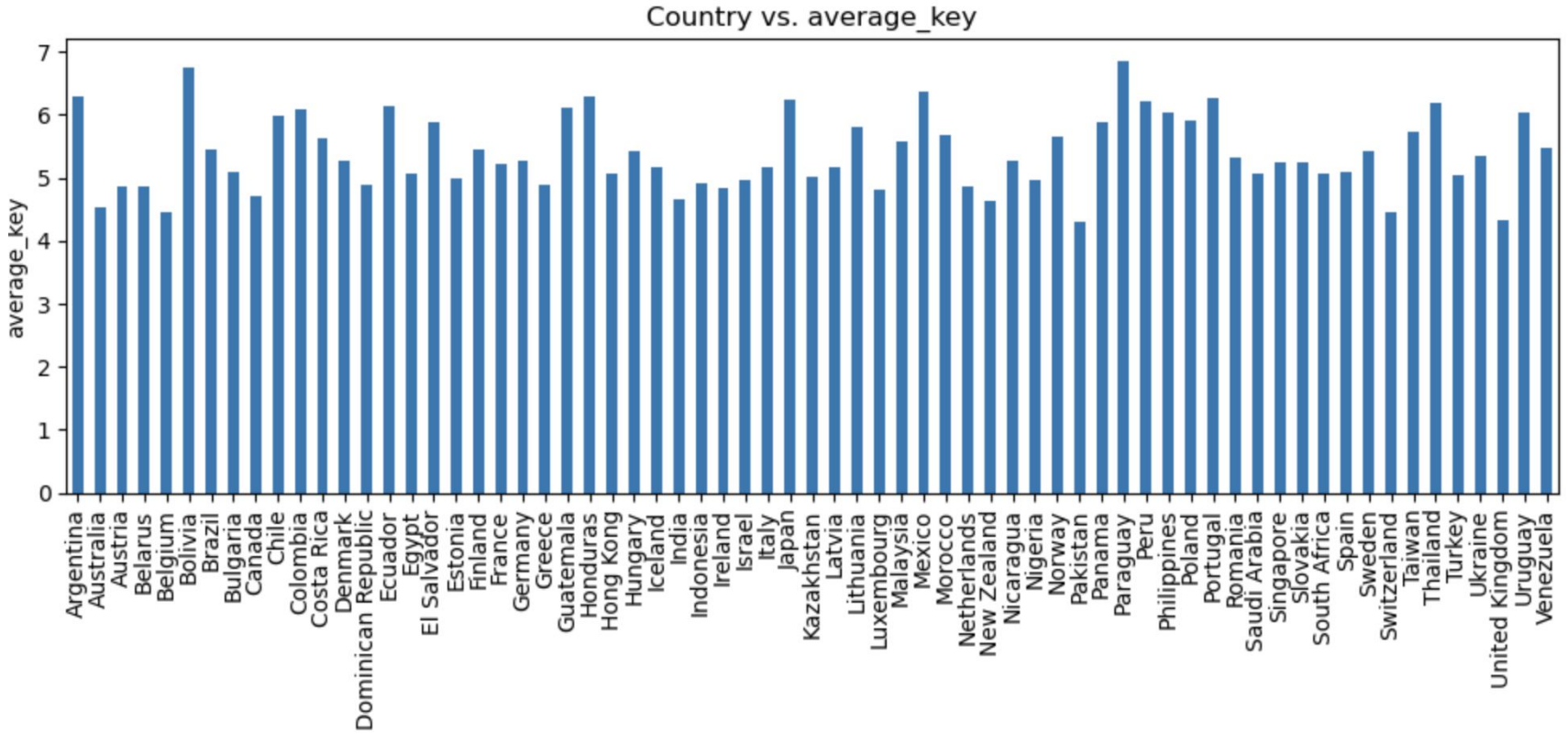


Country analysis

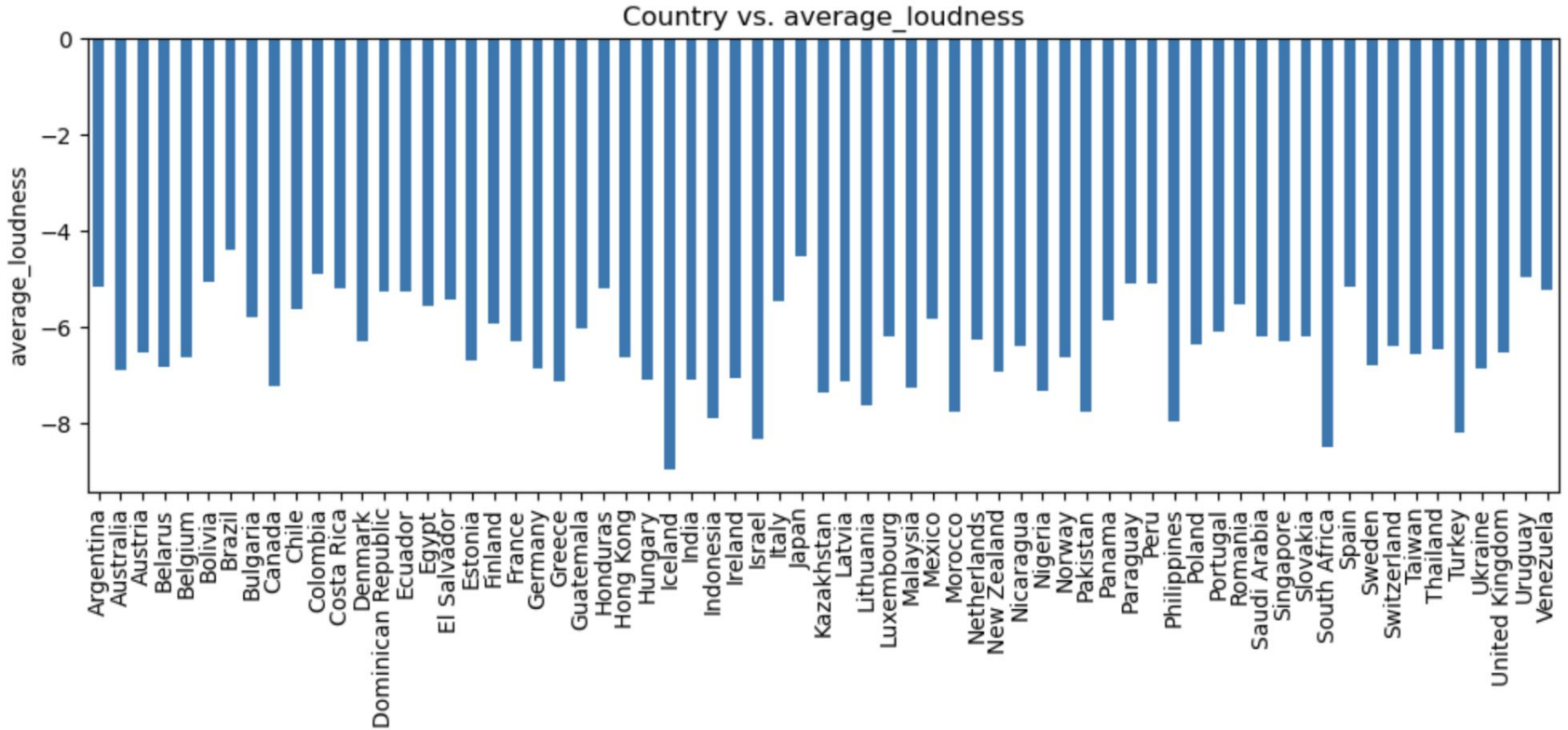
Analysis: Country music attribute - danceability



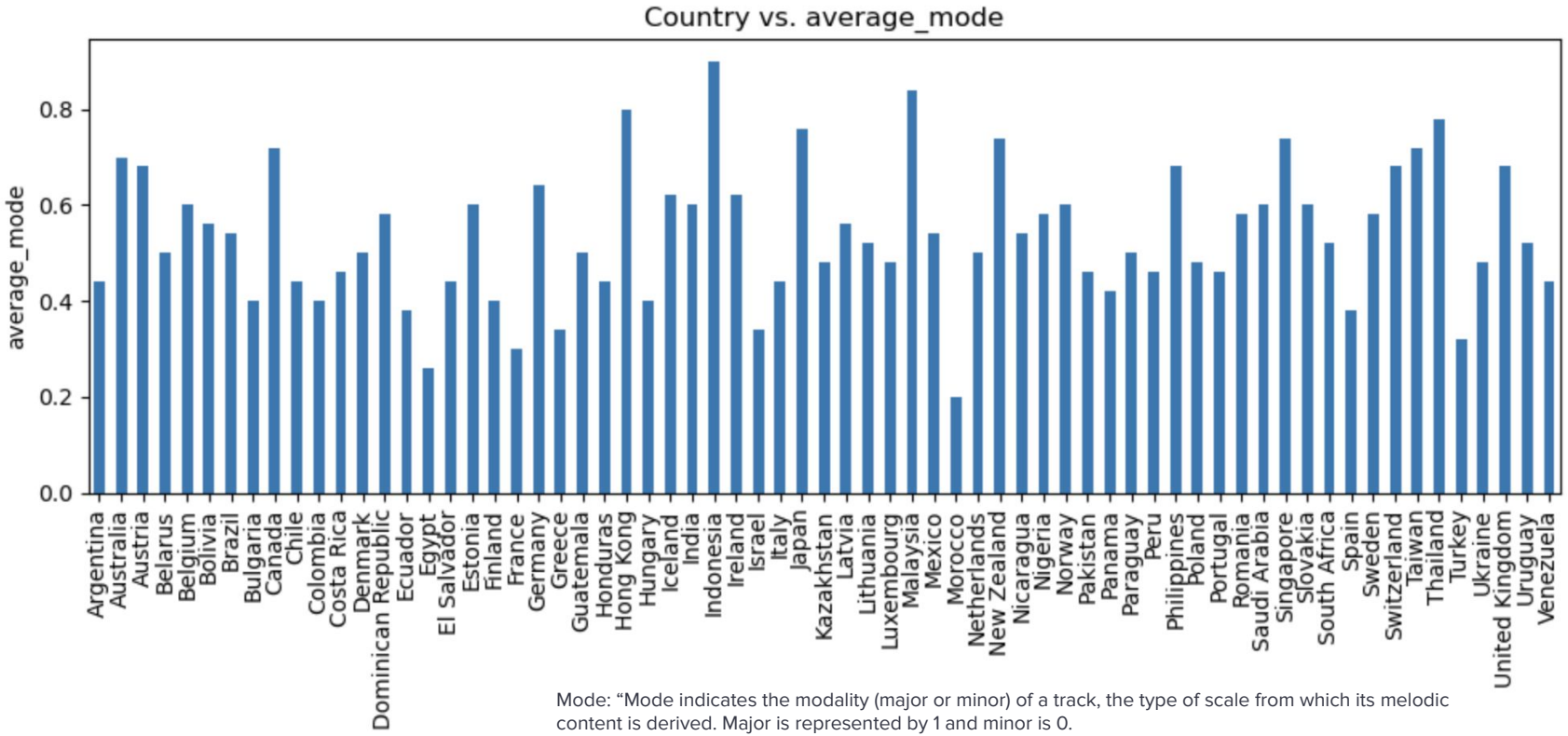
Analysis: Country music attribute - key



Analysis: Country music attribute - loudness

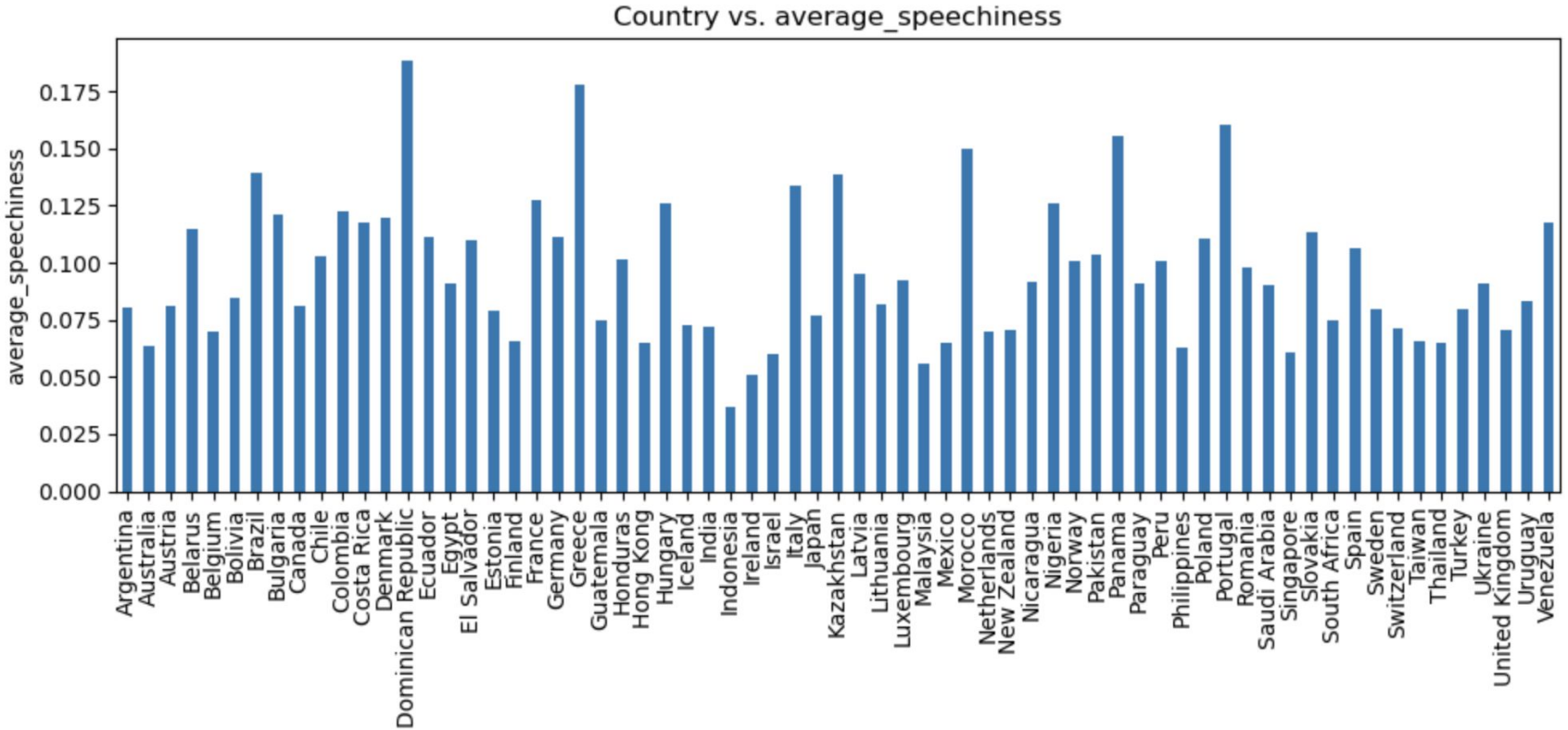


Analysis: Country music attribute - mode

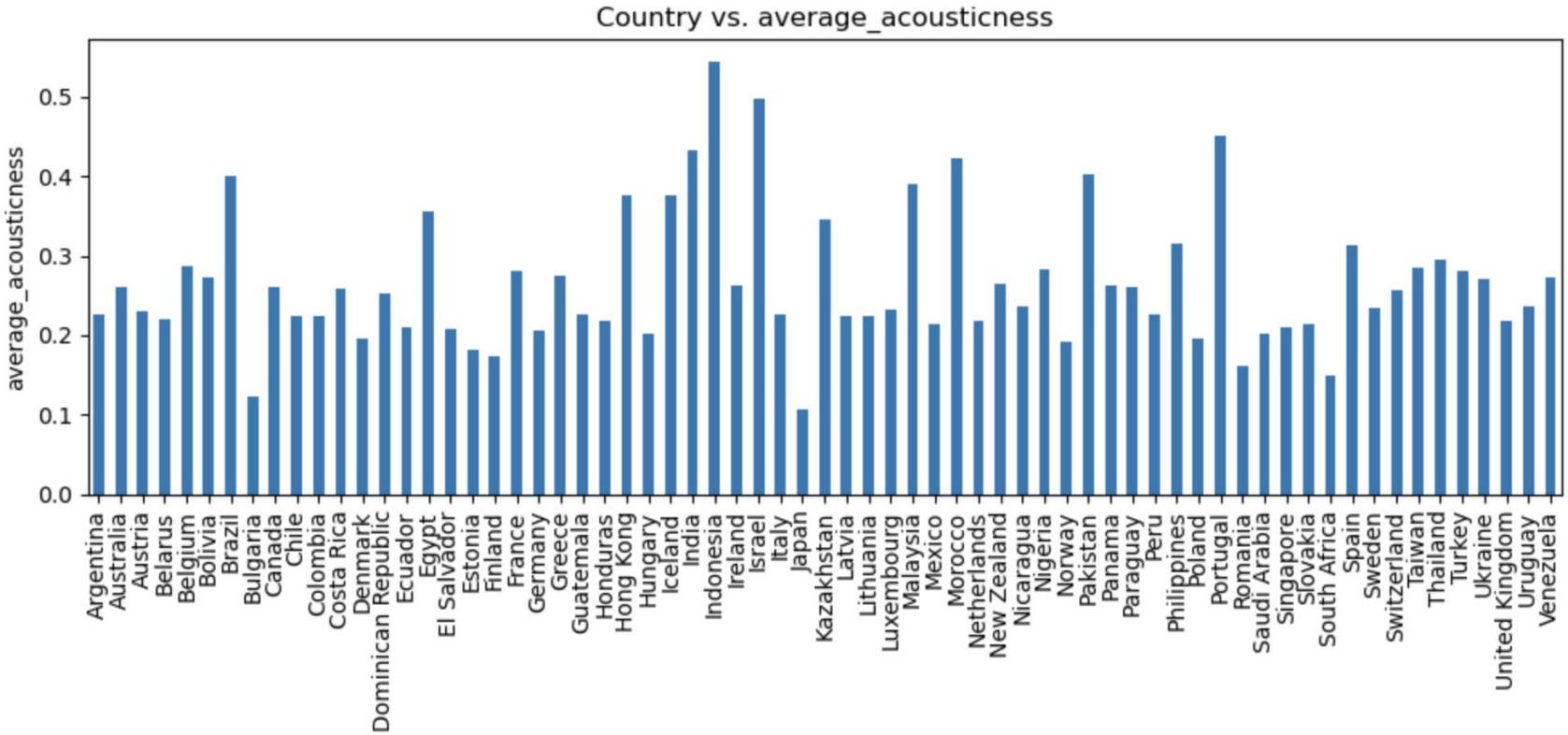


Mode: “Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0.
https://developer.spotify.com/documentation/web-api/reference/get-audio-features

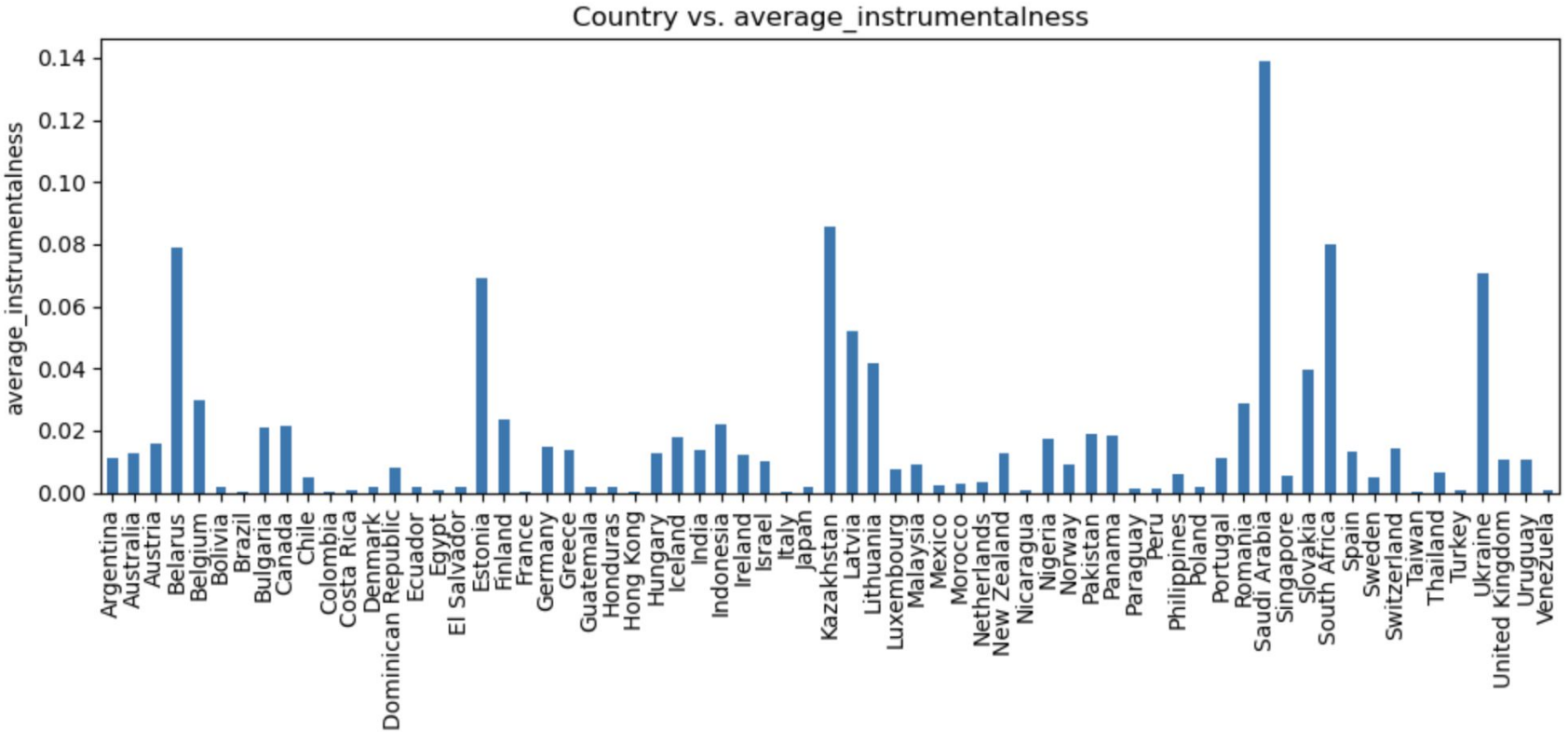
Analysis: Country music attribute - speechiness



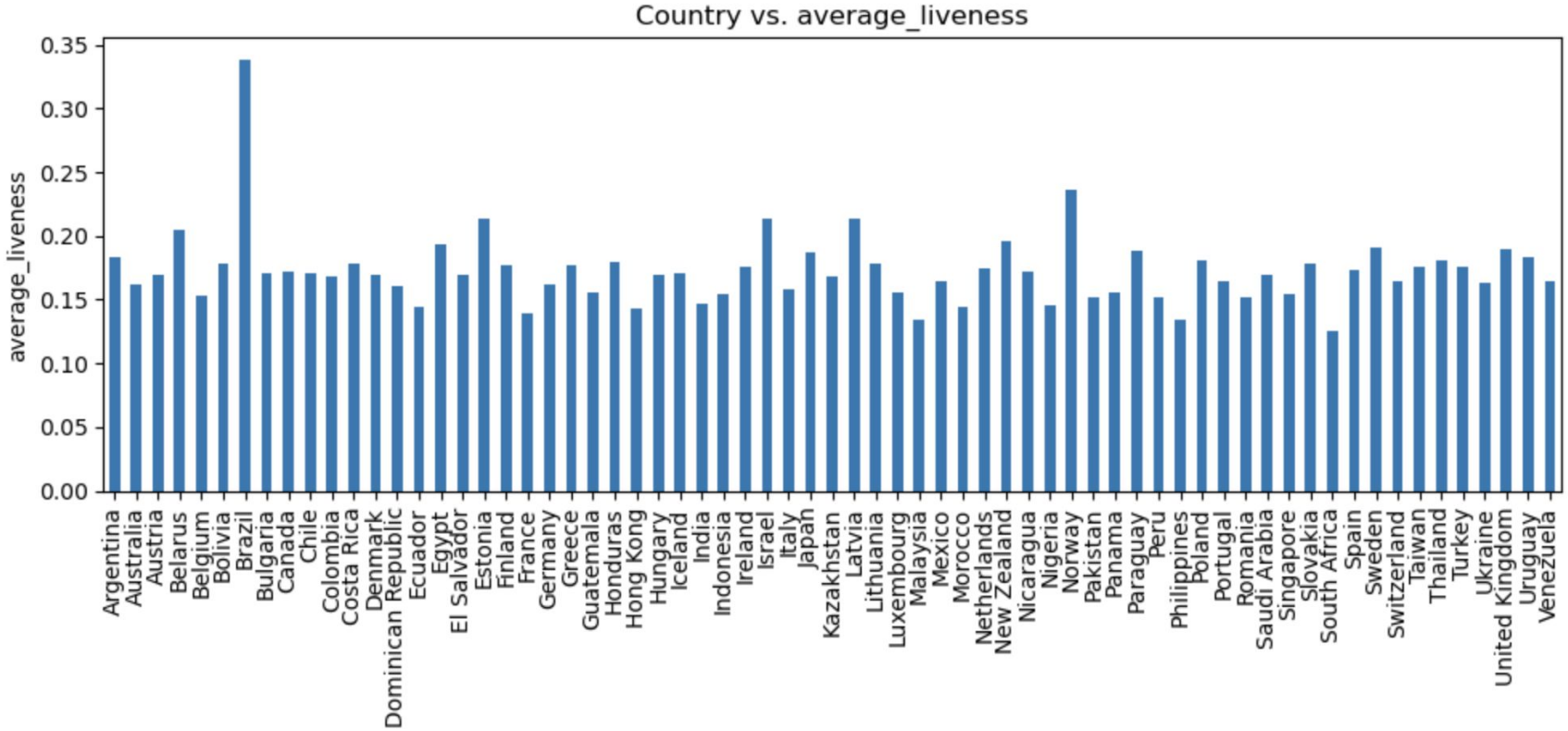
Analysis: Country music attribute - acousticness



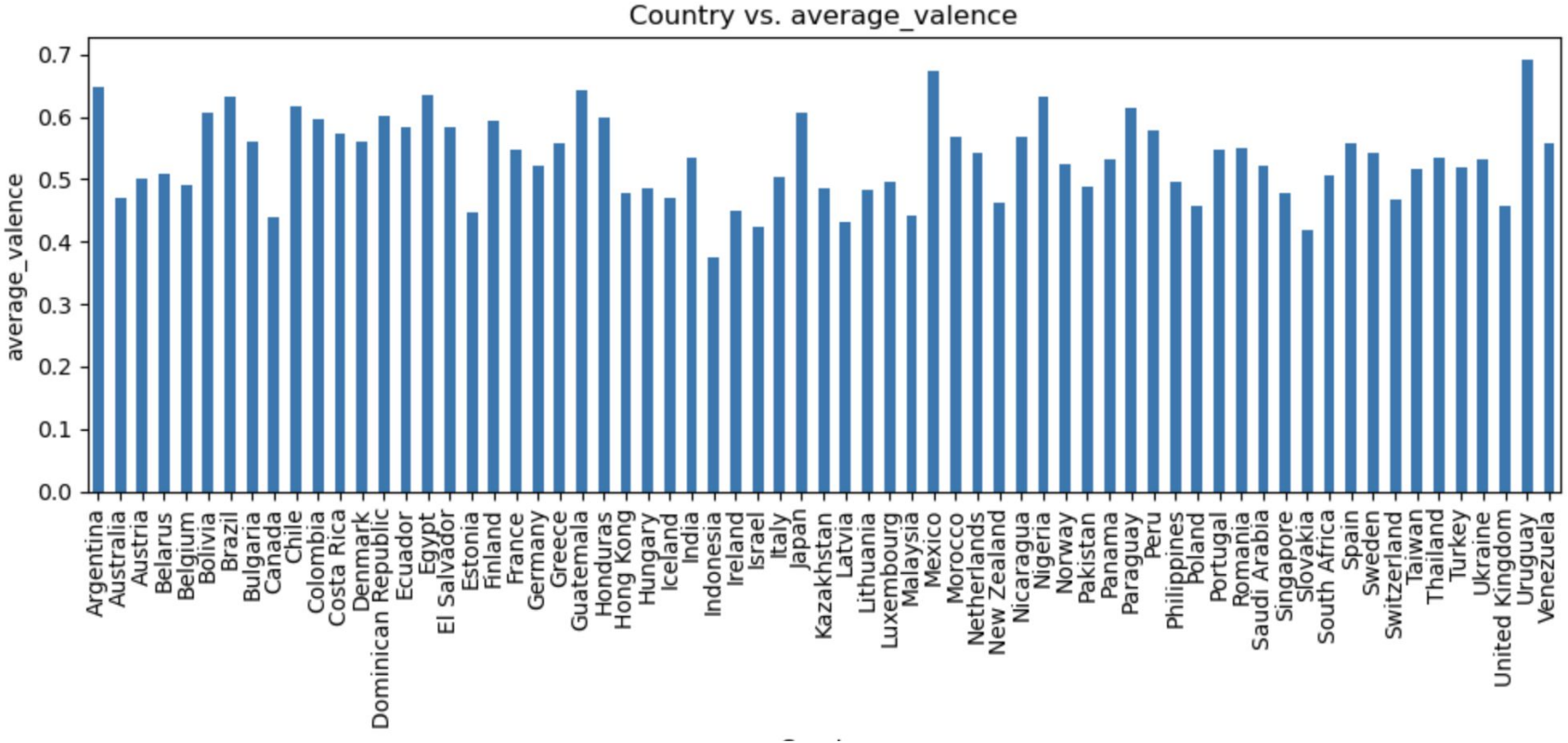
Analysis: Country music attribute - instrumentalness



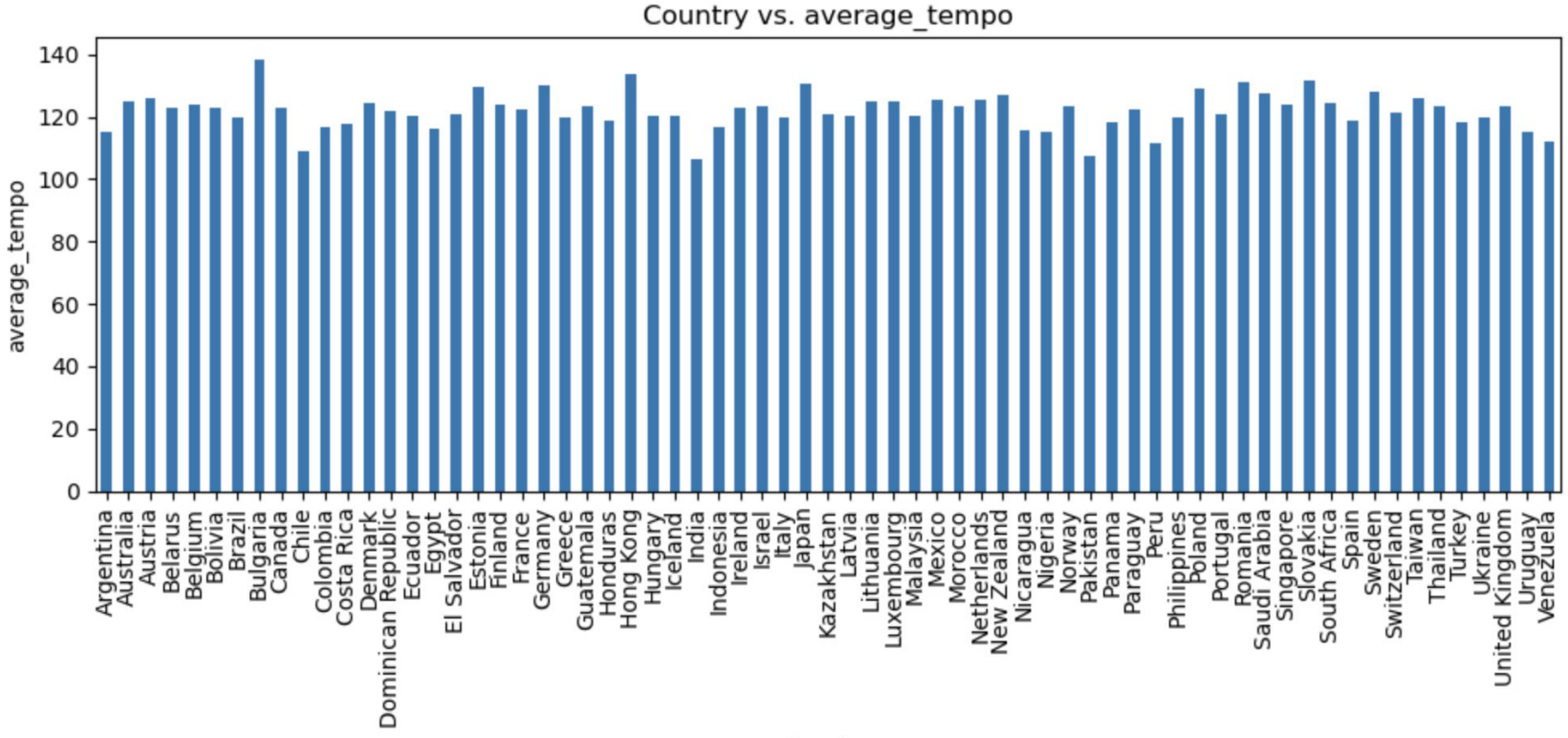
Analysis: Country music attribute - liveness



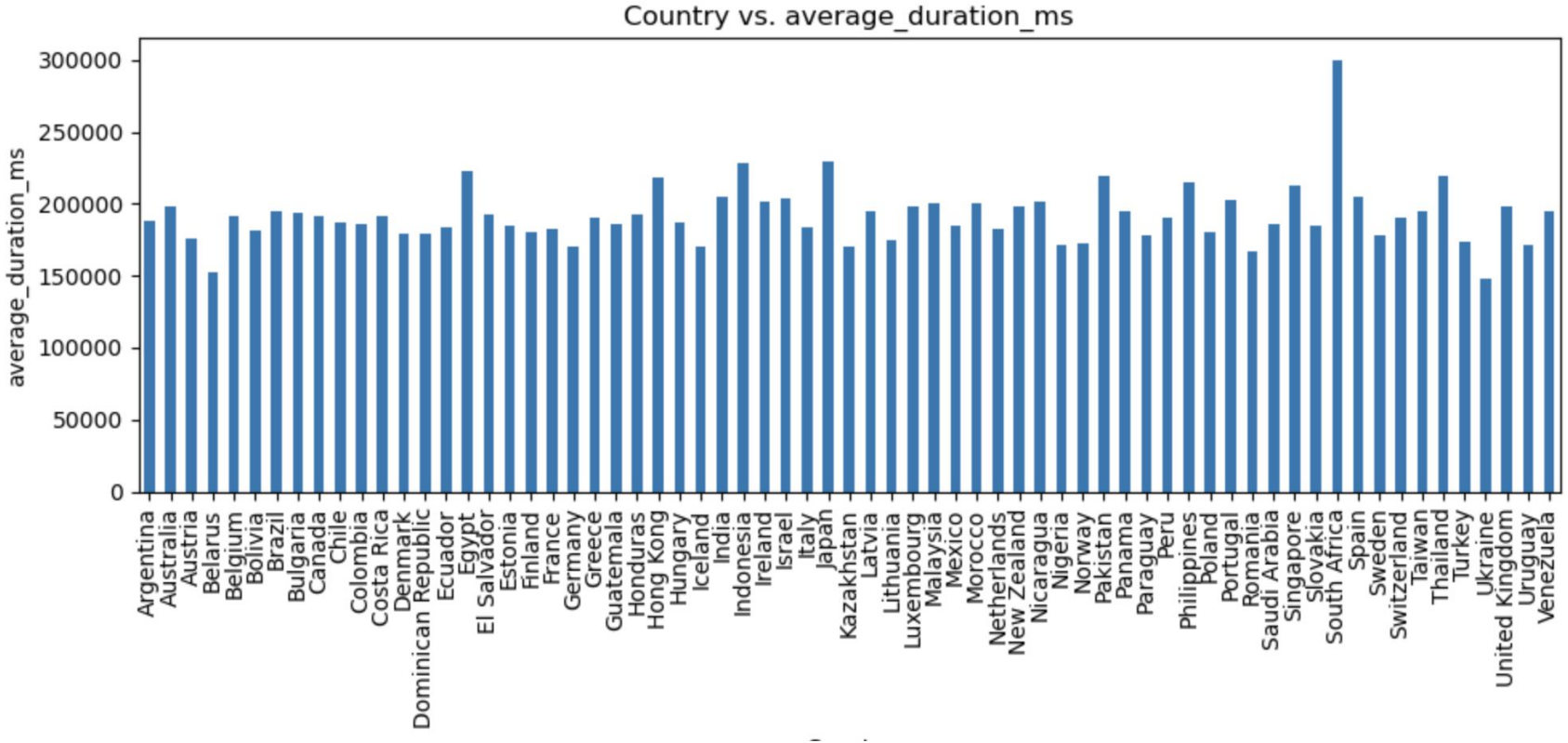
Analysis: Country music attribute - valence



Analysis: Country music attribute - tempo



Analysis: Country music attribute - duration (ms)



Analysis: Country music attribute - time signature