US Census Diversity vs. Music Genre Diversity

Anders Lundin, Ryan Horlick, Nathan Witt

https://github.com/panderz74/FinalProj206.git

Goals

Original Goals	• Extract insights, location data, and artists from Spotify • Create tailored travel playlists given a user's destination and travel plan
Achieved Goals	• Extract insights, artist information and more from Spotify and Billboard Top 100 (data from wikipedia) • Utilize Census API for US racial diversity and population statistics • **CONCLUSION:** Despite lacking causality between the two variables, interesting trends were independently found

Problems

- Our initial idea was more of a fun product than a true analysis of data
- Incapable of utilizing Google Maps API without a paid dashboard and subscription to Google Software
- No (quantifiable) data could be extracted from images via Pinterest or other potential album cover providers
- Taking 25 rows at a time from US Census API (Solved!)
- Identifying correct and useable data from US Census (Solved!)

Problems cont.

- Going from an interesting idea to an idea that was mathematically comparable.
- We had to completely rethink our project after developing access to the Spotify API, as the project did not have computable data that could be analyzed.
- How do we talk about racial diversity in a respectful way -- concerns about reflecting diversity through racial statistics when in relation to music.
- How to visualize a comparison of musical and statistical census data in a way in which makes sense.

documentation.txt

Each artist dictionary shows the frequency of appearances on the billboard 100 for each respective year.

Artist dictionary for 2000:
dict_items([("Destiny's Child", 3), ('Christina Aguilera', 3), ('Faith Hill', 2), ('Santana', 2), ('Savage Garden', 2), ('Lonestar', 2), ('Creed', 2), ('Aaliyah', 2), ('Sisqó', 2), ('Pink', 2), ("'N Sync", 2), ('Marc Anthony', 2), ('Jagged Edge', 2), ('Britney Spears', 2), ('Donell Jones', 2), ('Dr. Dre', 2), ('Joe', 1), ('Vertical Horizon', 1), ('Matchbox Twenty', 1), ('Toni Braxton', 1), ('3 Doors Down', 1), ('Madonna', 1), ('Janet', 1), ('Brian McKnight', 1), ('Montell Jordan', 1), ('Macy Gray', 1), ('Céline Dion', 1), ('Nelly', 1), ('Blaque', 1), ('Backstreet Boys', 1), ('Missy Elliott', 1), ('BBMak', 1), ('Sonique', 1), ('Nine Days', 1), ('Enrique Iglesias', 1), ('Ruff Endz', 1), ('Blink 182', 1), ('Third Eye Blind', 1), ('LeAnn Rimes', 1), ('Mariah Carey', 1), ('Whitney Houston', 1), ('Smash Mouth', 1), ('Eiffel 65', 1), ('Sting', 1), ('Eminem', 1), ('Next', 1), ('Everclear', 1), ('Jessica Simpson', 1), ('98 Degrees', 1), ('Filter', 1), ('Red Hot Chili Peppers', 1), ('Jay-Z', 1), ('Son by Four', 1), ('Avant', 1), ('Carl Thomas', 1), ('soulDecision', 1), ('Kid Rock', 1), ('Mystikal', 1), ('Erykah Badu', 1), ('Train', 1), ('DMX', 1), ('Mýa', 1), ('Jo Dee Messina', 1), ('Westlife', 1), ('Chad Brock', 1), ('George Strait', 1), ('Toby Keith', 1), ('Tim McGraw', 1), ('Goo Goo Dolls', 1), ('Da Brat', 1), ('Kandi', 1), ('Lee Ann Womack', 1), ('Alice Deejay', 1), ('Debelah Morgan', 1), ('Sammie', 1), ('Kevon Edmonds', 1), ('LFO', 1), ("Lil' Bow Wow", 1), ('Dixie Chicks', 1), ('Samantha Mumba', 1), ('Jennifer Lopez', 1), ('Mary Mary', 1)])
Artist dictionary for 2010:
dict_items([('Kesha', 4), ('Usher', 4), ('Lady Gaga', 4), ('The Black Eyed Peas', 4), ('B.o.B', 3), ('Jason Derulo', 3), ('Rihanna', 3), ('Drake', 3), ('Taylor Swift', 3), ('Lady Antebellum', 2), ('Katy Perry', 2), ('Eminem', 2), ('Taio Cruz', 2), ('Jay-Z', 2), ('Trey Songz', 2), ('Ludacris', 2), ('Timbaland', 2), ('Jay Sean', 2), ('OneRepublic', 2), ('Alicia Keys', 2), ('Train', 1), ('Enrique Iglesias', 1), ('Young Money', 1), ('Bruno Mars', 1), ('Mike Posner', 1), ('Travie McCoy', 1), ('Iyaz', 1), ('David Guetta', 1), ('The Script', 1), ('Owl City', 1), ('Nelly', 1), ('Far East Movement', 1), ('Michael Bublé', 1), ('Flo Rida', 1), ('La Roux', 1), ('Justin Bieber', 1), ('Adam Lambert', 1), ('Kris Allen', 1), ('Orianthi', 1), ('Neon Trees', 1), ('Maroon 5', 1), ('New Boyz', 1), ('Nicki Minaj', 1), ('Miley Cyrus', 1), ('Chris Brown', 1), ('Britney Spears', 1), ('Shontelle', 1), ('Boys Like Girls', 1), ('3OH!3', 1), ('Kelly Clarkson', 1), ('Selena Gomez & the Scene', 1), ('DJ Khaled', 1), ('Kevin Rudolf', 1), ('Sugarland', 1), ('Beyoncé', 1), ('Kings of Leon', 1), ('Carrie Underwood', 1), ('Sean Kingston', 1), ('Lil Wayne', 1), ('Miranda Lambert', 1), ('The Band Perry', 1), ('Paramore', 1), ('Sara Bareilles', 1), ('Daughtry', 1), ('Uncle Kracker', 1), ('Cali Swag District', 1), ('Jerrod Niemann', 1)])
Artist dictionary for 2020:
dict_items([('Lil Baby', 5), ('Pop Smoke', 4), ('Juice Wrld', 3), ('Luke Combs', 3), ('The Weeknd', 2), ('Roddy Ricch', 2), ('Dua Lipa', 2), ('DaBaby', 2), ('Harry Styles', 2), ('Lewis Capaldi', 2), ('Doja Cat', 2), ('Justin Bieber', 2), ('Billie Eilish', 2), ('Lizzo', 2), ('Drake', 2), ('Morgan Wallen', 2), ('Chris Brown', 2), ('Travis Scott', 2), ('Rod Wave', 2), ('Kane Brown', 2), ('Post Malone', 1), ('Future', 1), ('Maroon 5', 1), ('Maren Morris', 1), ('Gabby Barrett', 1), ('Jack Harlow', 1), ('Tones and I', 1), ('Megan Thee Stallion', 1), ('Arizona Zervas', 1), ('Imanbek', 1), ('Trevor Daniel', 1), ('Dan + Shay', 1), ('Cardi B', 1), ('Mustard', 1), ('Blackbear', 1), ('Lil Mosey', 1), ('Selena Gomez', 1), ('Jawsh 685', 1), ('Camila Cabello', 1), ('BTS', 1), ('Powfu', 1), ('Shawn Mendes', 1), ('24kGoldn', 1), ('Lady Gaga', 1), ('Black Eyed Peas', 1), ('Blake Shelton', 1), ('Shaed', 1), ('JP Saxe', 1), ('Old Dominion', 1), ('Jason Aldean', 1), ('Surfaces', 1), ('Eminem', 1), ('StaySolidRocky', 1), ('Maddie & Tae', 1), ('DJ Khaled', 1), ('Mariah Carey', 1), ('Lee Brice', 1), ('Sam Hunt', 1), ('Luke Bryan', 1), ('Lil Nas X', 1), ('Young Thug', 1), ('Carly Pearce', 1), ('YNW Melly', 1), ('Jonas Brothers', 1), ('Ariana Grande', 1), ('Halsey', 1), ('Benee', 1), ('Miranda Lambert', 1), ('Jhené Aiko', 1), ('Surf Mesa', 1), ('Moneybagg Yo', 1), ('H.E.R.', 1), ('NLE Choppa', 1)])

Each summary list gives a dictionary of genres and their frequency in the top 100 that year, along with the average danceability, energy, liveness, and tempo of the songs on the chart. Finally, dictionaries of how many times an artist appeared on the billboard at x appearances is calculated (if key is 3, and value is 1, one artist appeared three times on the billboard).

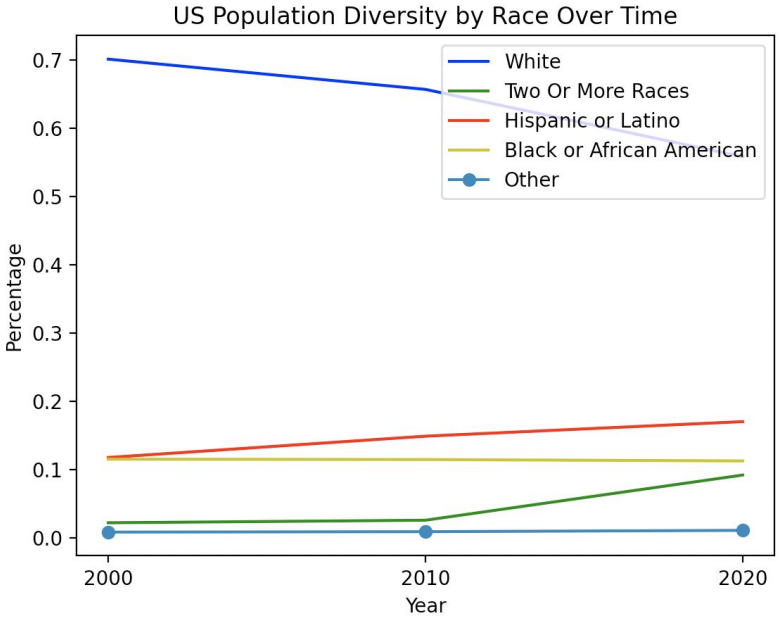
Summary of data for 2000:
Genre frequencies: {'dance pop': 40, 'urban contemporary': 35, 'r&b': 29, 'hip pop': 26, 'contemporary r&b': 22, 'pop': 18, 'hip hop': 18, 'pop rock': 14, 'country': 12, 'neo soul': 12, 'pop rap': 12, 'boy band': 11, 'country road': 10, 'rap': 10, 'contemporary country': 9, 'post-grunge': 9, 'quiet storm': 9, 'rock': 8, 'post-teen pop': 8, 'country dawn': 6, 'neo mellow': 6, 'europop': 6, 'alternative metal': 6, 'alternative rock': 6, 'nu metal': 5, 'southern hip hop': 5, 'gangster rap': 5, 'latin': 4, 'latin pop': 4, 'permanent wave': 4, 'girl group': 3, 'tropical': 3, 'eurodance': 3, 'pop r&b': 3, 'blues rock': 2, 'classic rock': 2, 'mexican classic rock': 2, 'australian pop': 2, 'modern salsa': 2, 'salsa': 2, 'new jack swing': 2, 'canadian pop': 2, 'atl hip hop': 2, 'bubblegum dance': 2, 'east coast hip hop': 2, 'g funk': 2, 'west coast rap': 2, 'soul': 1, 'st louis rap': 1, 'virginia hip hop': 1, 'mexican pop': 1, 'pop punk': 1, 'punk': 1, 'socal pop punk': 1, 'italian adult pop': 1, 'italo dance': 1, 'soft rock': 1, 'sophisti-pop': 1, 'detroit hip hop': 1, 'minneapolis sound': 1, 'grunge': 1, 'industrial metal': 1, 'industrial rock': 1, 'modern rock': 1, 'rap rock': 1, 'funk metal': 1, 'funk rock': 1, 'deep latin christian': 1, 'puerto rican pop': 1, 'crunk': 1, 'dirty south rap': 1, 'new orleans rap': 1, 'trap': 1, 'afrofuturism': 1, 'hardcore hip hop': 1, 'country rock': 1, 'oklahoma country': 1, 'chicago rap': 1, 'gospel': 1, 'gospel r&b': 1}
Average danceability: 0.6828399999999998
Average energy: 0.71325
Average liveness: 0.14298300000000003
Average tempo: 118.45304999999999
Frequency of appearances for each artist: {3: 2, 2: 14, 1: 66}

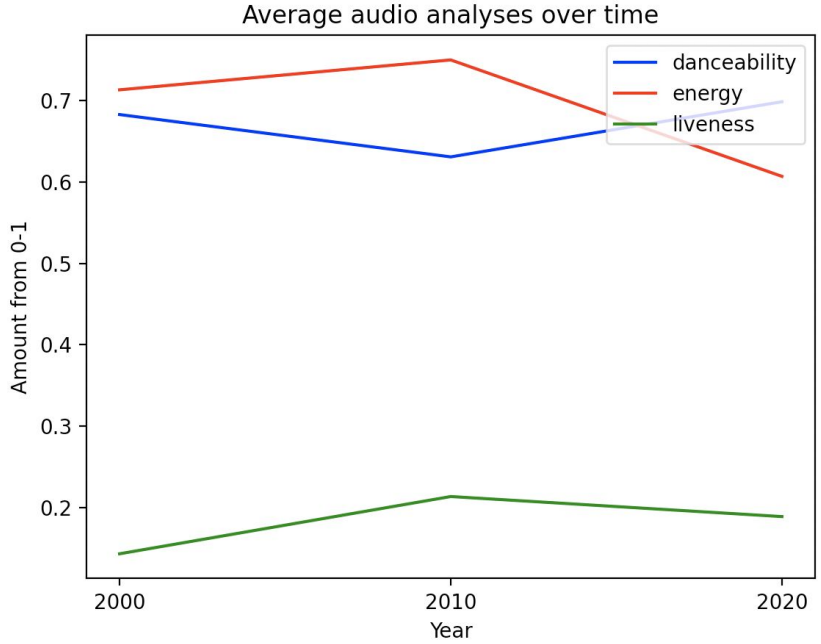
documentation.txt

```
Summary of data for 2010:
Genre frequencies: {'dance pop': 64, 'pop': 61, 'pop rap': 37, 'post-teen pop': 25, 'urban contemporary': 18, 'hip hop': 17, 'rap': 17, 'pop rock': 12, 'neo mellow': 11,
'southern hip hop': 11, 'r&b': 10, 'atl hip hop': 9, 'country': 8, 'country road': 8, 'contemporary country': 7, 'trap': 7, 'hip pop': 7, 'electropop': 6, 'country dawn': 6,
'art pop': 5, 'contemporary r&b': 4, 'south carolina hip hop': 4, 'barbadian pop': 4, 'modern rock': 4, 'canadian pop': 4, 'country pop': 3, 'dirty south rap': 3, 'edm': 3,
'canadian hip hop': 3, 'toronto rap': 3, 'miami hip hop': 3, 'pop punk': 3, 'modern country rock': 3, 'detroit hip hop': 2, 'east coast hip hop': 2, 'old school atlanta hip
hop': 2, 'idol': 2, 'acoustic pop': 2, 'piano rock': 2, 'candy pop': 2, 'rock': 2, 'neo soul': 2, 'latin': 1, 'latin pop': 1, 'mexican pop': 1, 'gangster rap': 1, 'big room':
1, 'pop dance': 1, 'celtic rock': 1, 'indietronica': 1, 'st louis rap': 1, 'asian american hip hop': 1, 'adult standards': 1, 'jazz pop': 1, 'lounge': 1, 'alternative dance':
1, 'electronica': 1, 'neo-synthpop': 1, 'new rave': 1, 'modern alternative rock': 1, 'queens hip hop': 1, 'neon pop punk': 1, 'electropowerpop': 1, 'talent show': 1, 'viral
pop': 1, 'oklahoma country': 1, 'new orleans rap': 1, 'pixie': 1, 'pop emo': 1, 'lilith': 1, 'alternative metal': 1, 'post-grunge': 1}
Average danceability: 0.63083
Average energy: 0.7500000000000001
Average liveness: 0.21334400000000003
Average tempo: 118.83237000000003
Frequency of appearances for each artist: {4: 4, 3: 5, 2: 11, 1: 47}


Summary of data for 2020:
Genre frequencies: {'pop': 42, 'rap': 26, 'dance pop': 21, 'contemporary country': 19, 'trap': 15, 'melodic rap': 13, 'pop rap': 13, 'country': 13, 'hip hop': 8, 'country
road': 8, 'canadian pop': 7, 'atl hip hop': 7, 'atl trap': 6, 'uk pop': 4, 'southern hip hop': 4, 'country pop': 4, 'electropop': 4, 'r&b': 4, 'brooklyn drill': 4, 'canadian
contemporary r&b': 3, 'trap queen': 3, 'art pop': 3, 'alternative r&b': 3, 'post-teen pop': 3, 'chicago rap': 3, 'north carolina hip hop': 2, 'pop rock': 2, 'slap house': 2,
'cali rap': 2, 'escape room': 2, 'minnesota hip hop': 2, 'canadian hip hop': 2, 'toronto rap': 2, 'nz pop': 2, 'florida rap': 2, 'alt z': 2, 'urban contemporary': 2, 'black
americana': 2, 'dfw rap': 1, 'texas country': 1, 'deep underground hip hop': 1, 'kentucky hip hop': 1, 'australian pop': 1, 'houston rap': 1, 'rhode island rap': 1, 'rap
conscient': 1, 'vapor trap': 1, 'k-pop': 1, 'k-pop boy group': 1, 'sad rap': 1, 'viral pop': 1, 'oklahoma country': 1, 'bedroom soul': 1, 'detroit hip hop': 1, 'miami hip hop':
1, 'modern country rock': 1, 'lgbtq+ hip hop': 1, 'gangster rap': 1, 'boy band': 1, 'etherpop': 1, 'indie poptimism': 1, 'country dawn': 1, 'tropical house': 1, 'memphis hip
hop': 1, 'tennessee hip hop': 1}
Average danceability: 0.6987000000000003
Average energy: 0.6069599999999999
Average liveness: 0.18869200000000003
Average tempo: 119.46796999999998
Frequency of appearances for each artist: {5: 1, 4: 1, 3: 2, 2: 16, 1: 53}



Population Totals for 2000:
Total White Population: 208832192
Total African American Population: 34397222
Total American Indian Alaskan Population: 2202726
Total Asian Population: 10196231
Total Hawaiian and Other Pacific Islander Population: 396463
Total Hispanic / Latino Population: 35126514
Total Two or More Races Population: 6670243


Population Totals for 2010:
Total White Population: 220846420
Total African American Population: 38651675
Total American Indian Alaskan Population: 2610561
Total Asian Population: 14609176
Total Hawaiian and Other Pacific Islander Population: 535644
Total Hispanic / Latino Population: 50145587
Total Two or More Races Population: 8787752


Population Totals for 2020:
Total White Population: 201762388
Total African American Population: 40814239
Total American Indian Alaskan Population: 3394344
Total Asian Population: 19795100
Total Hawaiian and Other Pacific Islander Population: 681358
Total Hispanic / Latino Population: 61608113
Total Two or More Races Population: 33340785
```

```
Population Totals for 2020:
Total White Population: 201762388
Total African American Population: 40814239
Total American Indian Alaskan Population: 3394344
Total Asian Population: 19795100
Total Hawaiian and Other Pacific Islander Population: 681358
Total Hispanic / Latino Population: 61608113
Total Two or More Races Population: 33340785


Total percent makeup of each race by year:
White 2000: 0.7011989671359992
White 2010: 0.6569157686924754
White 2020: 0.5582856629309351
African American 2000: 0.11549606556228491
African American 2010: 0.11497082358806962
African American 2020: 0.11293484728747671
American Indian Alaska 2000: 0.007396125957838967
American Indian Alaska 2010: 0.00776520935242508
American Indian Alaska 2020: 0.00939230353605669
Asian 2000: 0.03423603697020073
Asian 2010: 0.04345552933121425
Asian 2020: 0.05477393797640893
Hawaiian / Other Pacific Islanders 2000: 0.0013312097308619912
Hawaiian / Other Pacific Islanders 2010: 0.001593292705426297
Hawaiian / Other Pacific Islanders 2020: 0.001885348436316565
Hispanic / Latino 2000: 0.11794482019270389
Hispanic / Latino 2010: 0.14915988599969335
Hispanic / Latino 2020: 0.1704724381440656
Two or More Races 2000: 0.0223967744501103
Two or More Races 2010: 0.026139490330696045
Two or More Races 2020: 0.09225546168874041
```

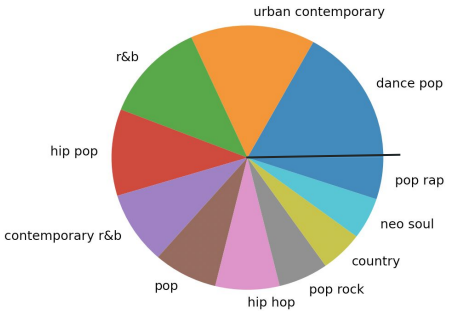
Visualizations



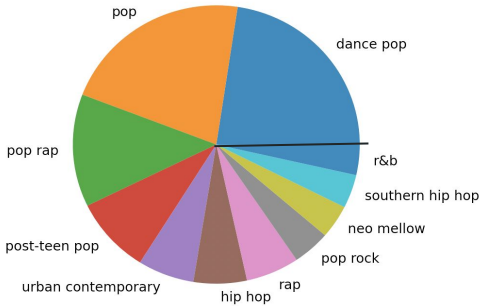


Visualizations (cont.)

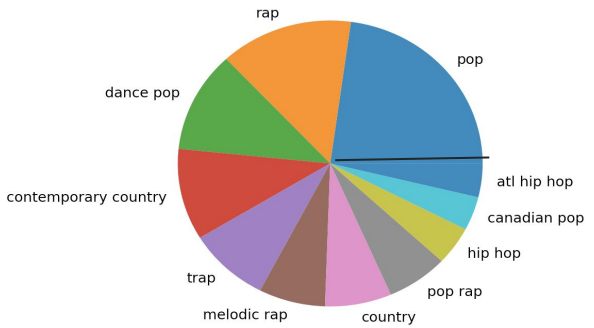
% make up of top 10 genres in 2000
Total genres: 81, Total % makeup of not shown genres: 46%



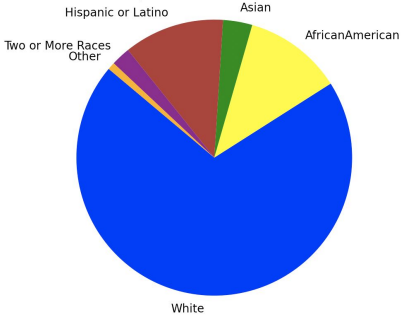
% make up of top 10 genres in 2010
Total genres: 72, Total % makeup of not shown genres: 35%



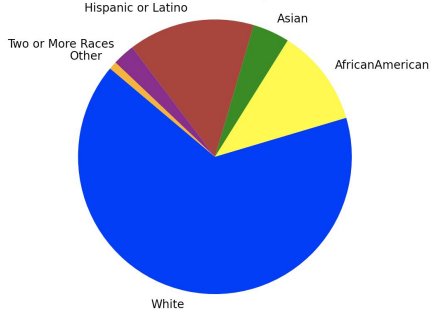
% make up of top 10 genres in 2020
Total genres: 65, Total % makeup of not shown genres: 36%



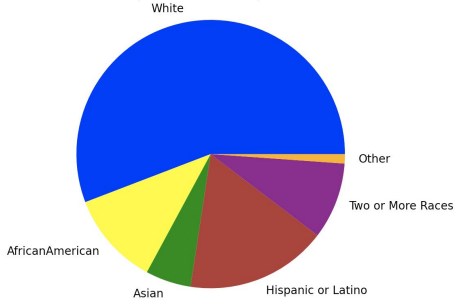
US Population Diversity by Race in 2000
Other = PacificIslander, American Indian, Alaska & Hawaiian Native



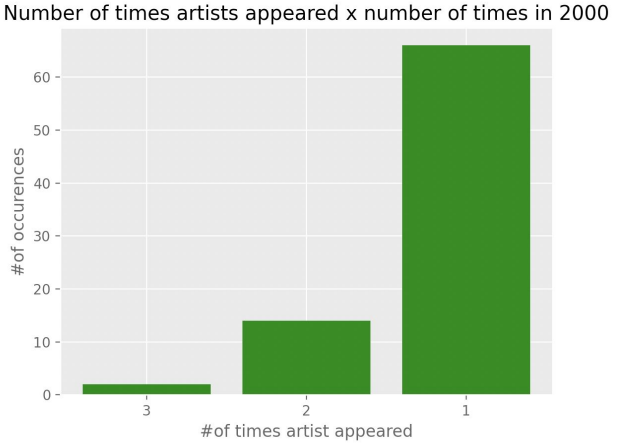
US Population Diversity by Race in 2010
Other = PacificIslander, American Indian, Alaska & Hawaiian Native

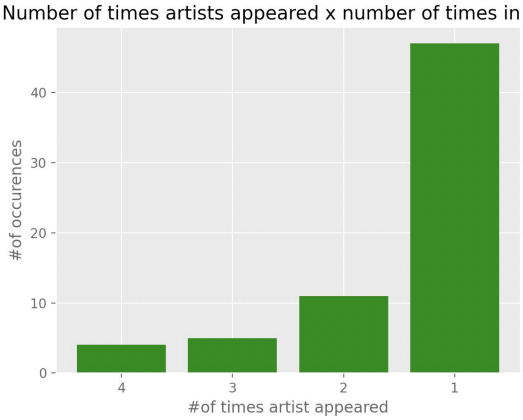


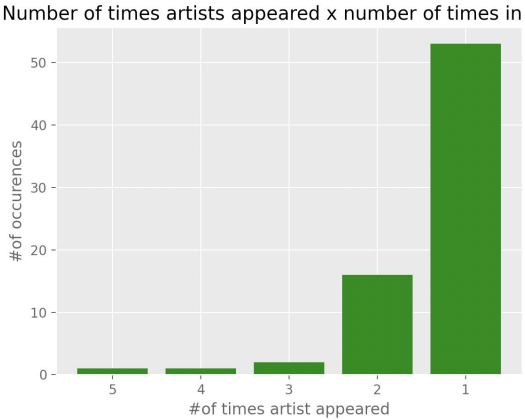
US Population Diversity by Race in 2020
Other = PacificIslander, American Indian, Alaska & Hawaiian Native



Visualizations (cont.)







Instructions: How to Run the Code

- SpotifyCensus.py must be ran 5 times, 4 to get the information into the database, and on the fifth run the code will display our visualizations.
- Census.py must be ran 3 times, the first two will get all fifty states data and the third will get the visualizations to display.

Code Documentation: What Each Function Does

SPOTIFY CENSUS:

- getsongs(pl): this function takes the id from each playlist dictionary and returns a list of ids.
- get_info(d, limit = 25, offset = 0): this function takes a dictionary of artists and song names, and searches spotify for them using spotipy.search(), returning a list of dictionaries including each song id(to look up the song on spotify, id is necessary), song name, artist name, and a list of genres that the song is in.

Code Documentation: What Each Function Does

SPOTIFY CENSUS cont. :

- get_audio_info(li): takes a list of dictionaries with from get_info() and returns the audio features of each track using spotipy.audio_features() function.
- scrape_top_music(year): takes a year as input and scrapes wikipedia using beautiful soup for the top songs and artists on the hot 100 billboard list for that year. It returns a dictionary of song: artist for that year.

Code Documentation: What Each Function Does

SPOTIFY CENSUS cont. :

- wiki_tables(): connects to the charts database and writes the wikipedia billboard100 information into the database. It creates three tables, one for each of the three called years.
- spotify_tables(count2000): inputs the count of how much data is already in the table, in order to only write 25 rows at a time. The function creates tables for each year with rows containing the id, track name, artist name, and genres for each track. It uses the wikipedia information and get_info() to accurately retrieve the information for the billboard100 through spotify.

Code Documentation: What Each Function Does

SPOTIFY CENSUS cont. :

- Artist_frequency(): this function makes a dictionary of how frequently each artist is on the top 100 for each year. Example format is “Juice WRLD: 3” for each artist. It then writes these calculations to a text file called “calculations”.

Code Documentation: What Each Function Does

SPOTIFY CENSUS cont. :

- year_analysis(): this function works to summarize all of the information regarding the different analyses for each of the three years. It joins the billboard data and audio analysis data for each year on the track id, and calculates the frequency of appearances for each genre into a dictionary, the average danceability, energy, liveliness, and tempo for each year’s songs, and a dictionary of how frequently an artist appears on the billboard x amount of times for that year (if key is 3, and value is 1, one artist appeared three times on the billboard), all before adding each to a summary list for that year. It then appends these summaries into the calculations text file.

Code Documentation: What Each Function Does

SPOTIFY CENSUS cont. :

- audio_lines(sumli): using the summary data returned by year_analysis(), this function creates a line chart showing the change in average danceability, energy, and liveness from 2000 to 2010 to 2020.
- genre_pies(): using the summary data returned by year_analysis(), this function creates pie charts that display the spread of the top ten genres for each year, along with what percentage of the list is not in the top ten. This gave interesting insight, because it showed a trend of less genres being mentioned overall in more recent years.

Code Documentation: What Each Function Does

SPOTIFY CENSUS cont. :

- artist_bars(sumli): using the summary data returned by year_analysis(), this displays a bar chart of how many artists are mentioned x amount of times per top100 year.

Code Documentation: What Each Function Does

Census.py:

- writefile(): using individual state race population data, makes a list of dictionaries containing the sum total of each race within each year and writes those totals into a document detailing calculations. Finally, it returns that dictionary.

Code Documentation: What Each Function Does

Census.py:

- writepercents(totli): takes in a dictionary created by writefile and sums the total population for each year, then calculates the percentage of each race given the total population and puts them in a list of percentages. Finally, writes a document with each percentage, for each year, for each race, before returning the list of race percentages.

Code Documentation: What Each Function Does

Census.py:

- main(): Creates table of US Census State, Race and, Population data, and if the table isn’t at 50, it loops twice to fill the table and calls the APIS (this is in order to keep 25 rows at a time).
- viz(): takes in a list after writepercents() is ran on the output of writefile(), then constructs a line plot detailing changing percentages of diversity in America from 2000 to 2020 using the intake.
- viz2(): takes the same list as viz(), and using that list it creates a pie chart for each year showing the percent makeup of each race according to total population for that year.

Resources Used

Date	Issue Description	Location of Resource	Result (did it solve the issue?)
12.04	Acquiring racial diversity data and population statistics from 2000,2010,2020	US Census Decennial Data: 2000, 2010,2 020	Provided a statistical backend for racial data, analyses, and visualizations.
12.04	Given top 100 songs, identify song ID, artist, audio, and other Spotify variables to later quantitatively analyze.	Spotify, Spotipy API	Yes.
12.04	Identify top 100 songs in the years 2000, 2010, and 2020.	Wikipedia Billboard 100	Yes.
12/09	Struggling to extract information from US Census API Links	Youtube Video(s): https://www.youtube.com/watch?v=yHeXpXwm6g4&t=199s , https://www.youtube.com/watch?v=5gAxns_342o	Yes.
12.12	Creating Matplotlib visualizations	https://matplotlib.org/stable/index.html	Yes.
12.11	Extracting data through API 25 rows at a time.	https://stackoverflow.com/questions/54101781/how-can-i-use-python-requests-to-paginate-api-calls-with-offset-parameter?rq=1	No, did not work with Census API. Came up with own solution instead.
12.01	Basic Spotify structure, getting api keys and setting up imports	https://medium.com/@maxtingle/getting-started-with-spotifys-api-spotipy-197c3dc6353b https://www.section.io/engineering-education/spotify-python-part-1/	Gave basic structure through two resources, yes.
12.08	Getting song genres as it is not immediately obvious	https://stackoverflow.com/questions/61624487/extract-artist-genre-and-song-release-date-using-spotipy	Gave a workaround by searching through artist and album, yes
12.01	Basic Spotipy documentation	https://spotipy.readthedocs.io/en/2.19.0/#module-spotipy.oauth2	Yes.

Resources Used

12.01	Basic Spotipy documentation	https://spotipy.readthedocs.io/en/2.19.0/#module-spotipy.oauth2	Yes.
12.13	How to write to a file that already exists	https://www.guru99.com/reading-and-writing-files-in-python.html	Showed how to append data, yes.
12.12	How to create graph with multiple titles	https://stackoverflow.com/questions/1388450/giving-graphs-a-subtitle-in-matplotlib	Yes.
12.11	How to create pie chart and edit quantities	https://www.w3schools.com/python/matplotlib_pie_charts.asp	Yes.
12.11	How to add legend to visualization	https://stackoverflow.com/questions/19125722/adding-a-legend-to-pyplot-in-matplotlib-in-the-simplest-manner-possible	Yes.
12.11	How to make line plot	https://www.w3schools.com/python/matplotlib_line.asp	Yes.
12.10	How to make bar chart	https://benalexkeen.com/bar-charts-in-matplotlib/	Yes.
12.08	How to join tables in simplest manner	https://stackoverflow.com/questions/29882139/create-table-from-join-sql-server	Yes.
12.07	Reminder of basic Sqlite functions	https://appdividend.com/2020/10/14/how-to-create-sqlite-database-in-python/	Yes.
12.10	Converting string of list to list type	https://stackoverflow.com/questions/1894269/how-to-convert-string-representation-of-list-to-a-list	Yes.