1 Introduction

In 2018, two of my favorite bands – The Front Bottoms and Manchester Orchestra – released a song they collaborated on called "Allentown." In a statement to Noisey (Ross, 2018) – the music arm of Vice – Andy Hull of Manchester Orchestra recalled that the creation of this track started when Nate Hussey of All Get Out sent him the first four lines of the track. Andy Hull worked out the melody and music and shared it with Brian Sella of The Front Bottoms, who then helped develop the chorus.

This brings us to an interesting question: which band contributed most to the song?

To attempt to answer this question, we purchased all releases before "Allentown" except joint albums, live albums, and single releases contained in a full album or an Extended Play (EP), a release that is more than a single release but shorter than a full album. We have 42 tracks by All Get Out, 77 by Manchester Orchestra, and 61 by The Front Bottoms. This totals 180 tracks; there are 181, including "Allentown."

We aimed to use Essentia (Bogdanov et al., 2013) – an open-source program for music analysis, description, and synthesis – to create data about what each band's tracks "sound like." We had trouble installing the program onto Windows and Linux. However, Essentia provides precompiled versions of the program that you can run on individual tracks using the command line[1]. Instead of programming, we can call the executable file to process each track (a `.wav` file we have). In the previous lab, you wrote code that can be generalized to create a batch file that contains all 181 commands, which we can execute to process all the songs.

The Research and High-Performance Computing group on campus was able to install the Essentia environment on a virtual computer. There, I was able to reproduce the `streaming_music_extractor` calls using the most recent version. I was also able to interface with Essentia models (Alonso-Jiménez et al., 2020), which provide additional features (e.g., moods, contexts, instrumentation, etc.). Unlike the data from the `streaming_music_extractor` calls, the data from the Essentia models are output to a .csv file.

In the coding task below, I provide the details necessary for completing the task of cleaning these data given the 181 `.json` files and the `.csv` file with data from the Essentia models. This lab will enable you to practice (1) installing, loading, and learning to use libraries; (2) working with character objects; (3) coding `for()` loops; and (4) accessing elements of vectors and lists.

2 Lab Coding Task: Compile Data from Essentia

Step 0: Install the `stringr` (Wickham et al., 2019) and `jsonlite` (Ooms, 2014) packages for `R`.
Step 1: As an example, work with just the song Au Revoir (Adios) on the Talon Of The Hawk album by The Front Bottoms.

1. Create an object called `current.filename` and set it equal to the character string: The Front Bottoms- Talon Of The Hawk- Au Revoir (Adios).json.

2. Use the `str_split()` function to extract the artist, album, and track from the filename. Note that while you can do this manually for one file, we want to automate this process for all files later.
   Hint: Use `str_sub()` to remove the trailing `.json`.

3. Load the JSON file into `R` using the `fromJSON()` function. The resulting object is a large list.

4. Extract the overall loudness (`loudness_ebu128$integrated` saved as `overall_loudness`), spectral energy (`spectral_energy`), dissonance (`dissonance`), pitch salience (`pitch_salience`), tempo in beats per minute, (`bpm`), beat loudness (`beats_loudness`), danceability (`danceability`), and tuning frequency (`tuning_frequency`). You will find the documentation here rather helpful as you explore the list object.

[1] https://essentia.upf.edu/extractors/

Step 2: Load and clean the data from the Essentia models by completing Step 1 for all `.JSON` files in the `EsssentiaOutput` folder. Save the resulting vector of data for each `.JSON` file as a row in a data frame.
Step 3: Load and clean the data from the Essentia models by completing the following steps. Note you may find the documentation here to be helpful.

1. Load the `EssentiaModelOutput.csv` file.
2. Valence and arousal are estimated using three datasets: DEAM, emoMusic, and MuSe. Compute two new columns `valence` and `arousal` that are the average of the three estimates of each.
3. Aggressive, happy, party, relaxed, and sad moods are collected using two feature extractors: Discogs-EffNet and MSD-MusiCNN. Compute new columns `aggressive`, `happy`, `party`, `relaxed`, and `sad` by averaging the two extractors for each feature.
4. Acoustic and electric sound are collected using two feature extractors: Discogs-EffNet and MSD-MusiCNN. Compute new columns `acoustic` and `electric` by averaging the two extractors.
5. Instrumental (absence of voice) is collected using two feature extractors: Discogs-EffNet and MSD-MusiCNN. Compute a new column `instrumental` that is the average of the two extractors.
6. Timbre is collected using only the Discogs-EffNet extractor. Rename the `eff_timbre_bright` column to `timbreBright`.
7. Retain only the features created or renamed above and the `artist`, `album`, `track` columns.

Step 4: I used the collected lyrics for the tracks and ran them through a text analysis tool called LIWC, which provides features that describe thoughts, feelings, and personality traits based on the language used in lyrics by the artists. Load the data from LIWC and compile the full dataset.

1. Load the `LIWCOutput.csv` file.
2. Use the `merge()` function to merge the data from the data from the `streaming_music_extractor` calls, the Essentia models, and LIWC into one data frame.
   Hint 1: You may find that your resulting dataframe "creates" rows. I found this post which describes the issue as being duplicated observations in the `by` argument. Note that the last example in the documentation for the `merge()` function enables grouping by more than one column.
   Hint 2: You may find that your resulting dataframe "removes rows. In this case, you may find the `all.x` or `all.y` arguments to be helpful.
3. Rename the `function` column to `funct`, noting that using `function` as a column name can cause issues while coding. For example, you'll see that `R` has already renamed it to `function.`.

Step 5: Create two separate files.

1. Write a `.csv` file called `trainingdata.csv` that contains all tracks except "Allentown."
2. Write a `.csv` called `testingdata.csv` that contains just the track "Allentown."

3 Lab Coding Challenge: What Do the Data Tell Us?

The instructions in this lab coding task lead you through creating a complex dataset that merges music and lyrical analyses from multiple programs. Now, we have a lot of data to answer our research question. In the next lab, we will work with these data using `tidyverse` (Wickham et al., 2019), which will include summarizing the data in ways that provide preliminary information about which band had the largest impact on "Allentown." As a challenge, attempt to create 1-3 plots that give insight into possible answers to this question. As a starting point, you may find the documentation for `boxplot()` helpful. For a more attractive plot, you might find the Shiny app here to be helpful; note that you download the plots directly or copy the generating `R` code by clicking the "</> R code" button.

References

Alonso-Jiménez, P., Bogdanov, D., Pons, J., and Serra, X. (2020). Tensorflow audio models in Essentia. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 266–270. IEEE.

Bogdanov, D., Wack, N., Gómez Gutiérrez, E., Gulati, S., Boyer, H., Mayor, O., Roma Trepat, G., Salamon, J., Zapata González, J. R., Serra, X., et al. (2013). Essentia: An audio analysis library for music information retrieval. In *Britto A, Gouyon F, Dixon S, editors. 14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4-8; Curitiba, Brazil.[place unknown]: ISMIR; 2013. p. 493-8.* International Society for Music Information Retrieval (ISMIR).

Ooms, J. (2014). The jsonlite package: A practical and consistent mapping between JSON data and R objects. *arXiv:1403.2805 [stat.CO]*.

Ross, A. R. (2018). Manchester Orchestra and The Front Bottoms are finally together on "Allentown". https://www.vice.com/en/article/manchester-orchestra-and-the-front-bottoms-are-finally-together-on-allentown/.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.