Pitchfork Review Analysis

By Dan Godwin, Evan Moore, Michael Palmer, Greg Rowe, and Dan Szymanski

Motivation

- We initially recognized that we wanted to work with some sort of creative corpus such as lyrics, poetry, or movie/TV scripts.
- Once we identified the idea of working with a corpus of reviews of such works instead, we knew we had a project worth pursuing further.
- Motivating questions include:
    - How does word use changes in different genres of reviews, especially adjectives?
    - Do reviews with higher scores tend to have different language used?
    - What are the words identified with most with each genre of reviews?
    - How does word use change over time?
    - How does average score change over genre, and by year?

Initial Ideas

- We thought it would be interesting to scrape from many different review sites (both professional and user-based) to gather a corpus that had review source and professional status as another feature.
  - Sites considered included AllMusic, Pitchfork, Rolling Stone, SputnikMusic, and Metacritic
- In practice, this ended up being too difficult and some sites did not lend themselves well to being scraped, so we chose to focus just on Pitchfork instead.

Corpus Collection

- We used the BeautifulSoup library to scrape a corpus of roughly 20,000 Pitchfork reviews (consisting of the entire amount available on the site).
- Functions were created that allow a user to specify a start page and number of pages to scrape from Pitchfork's reviews archive, then have a tidy Pandas DataFrame of observations saved as a .csv file at the specified file path.
- Features include score, date, genre, album, artist, a "Best New" indicator, and the text of the review saved as a string.
- The R programming language was used to clean and format the data and perform exploratory data analysis (EDA).

Data Example

	X	album	artist	best	date	genre	review	score
1	1	A.M./Being There	Wilco	1	December 6 2017	Rock	Best new reissue 1 / 2 Albums Newly reissued an...	7.0
2	2	No Shame	Hopsin	0	December 6 2017	Rap	On his corrosive fifth album, the rapper takes aim...	3.5
3	3	Material Control	Glassjaw	0	December 6 2017	Rock	On their first album in 15 years, the Long Island p...	6.6
4	4	Weighing of the Heart	Nabihah Iqbal	0	December 6 2017	Pop/R&B	On her debut LP, British producer Nabihah Iqbal—f...	7.7
5	5	The Visitor	Neil Young / Promise of the Real	0	December 5 2017	Rock	While still pointedly political, Neil Young's latest wit...	6.7
6	6	Perfect Angel	Minnie Riperton	1	December 5 2017	Pop/R&B	Best new reissue A deluxe reissue of Minnie Riper...	9.0
7	7	Everyday Is Christmas	Sia	0	December 5 2017	Pop/R&B	Sia's shiny Christmas album feels inconsistent an...	5.8
8	8	Zaytown Sorority Class of 2017	Zaytoven	0	December 5 2017	Rap	The prolific Atlanta producer enlists 17 women for...	6.2
9	9	Songs of Experience	U2	0	December 4 2017	Rock	Years in the making, U2's 14th studio album finds...	5.3
10	10	Post Self	Godflesh	0	December 4 2017	Metal	The new LP from pioneering industrial band Godfl...	8.1
11	11	cybersex	blackbear	0	December 4 2017	Rap	With a long bench of guests from 2 Chainz to Ne-Y...	4.1
12	12	Endless Computer	Expander	0	December 4 2017	Metal	With their vicious debut LP, Austin quartet Expand...	7.8
13	13	Metal Machine Music	Lou Reed	0	December 3 2017	Rock	Lou Reed's 1975 album has been called one of th...	8.7
14	14	Master of Puppets	Metallica	1	December 2 2017	Metal	Best new reissue In 1986, Metallica released inar...	10.0
15	15	Oblivion	T-Pain	0	December 2 2017	Rap	T-Pain's first studio album in six years cashes in ...	6.7
16	16	Friday on Elm Street	Fabolous / Jadakiss	0	December 2 2017	Rap	On the New York rappers' collaboration, just about...	6.0
17	17	Loüm / Go Be Forgotten	Krallice	0	December 2 2017	Metal	1 / 2 Albums New York City's most consistent met...	7.6

Summary Statistics

Mean Score: 7.027

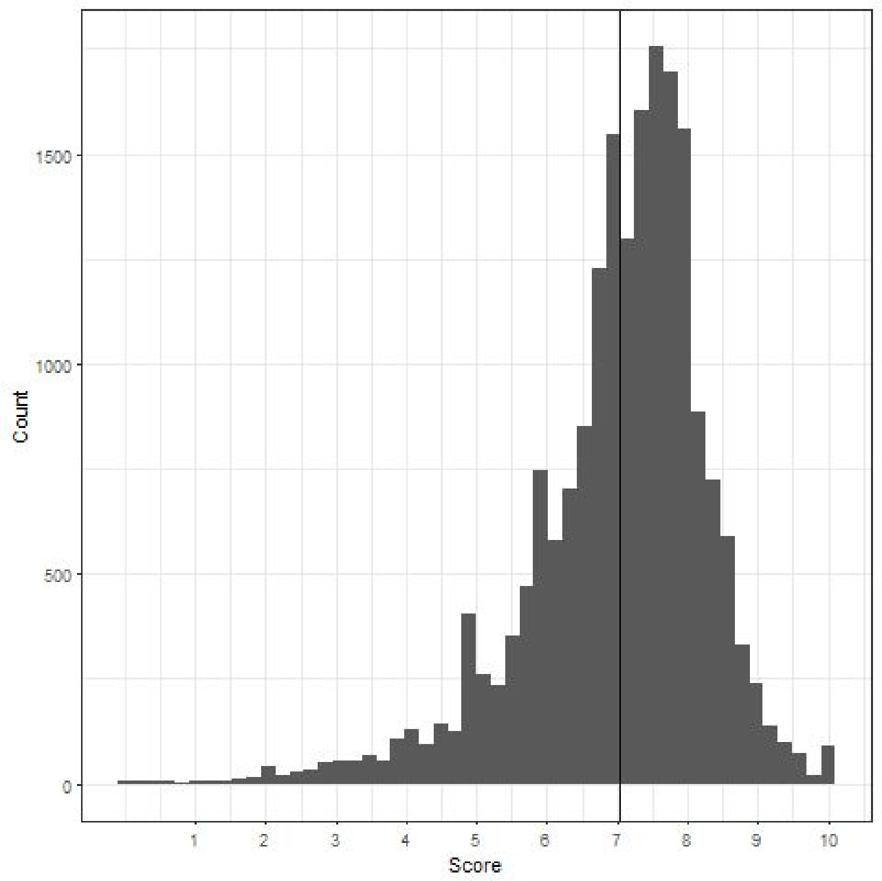
Percentage “Best”: 5.318%

Date Range:

1/5/1999 - 12/6/2017

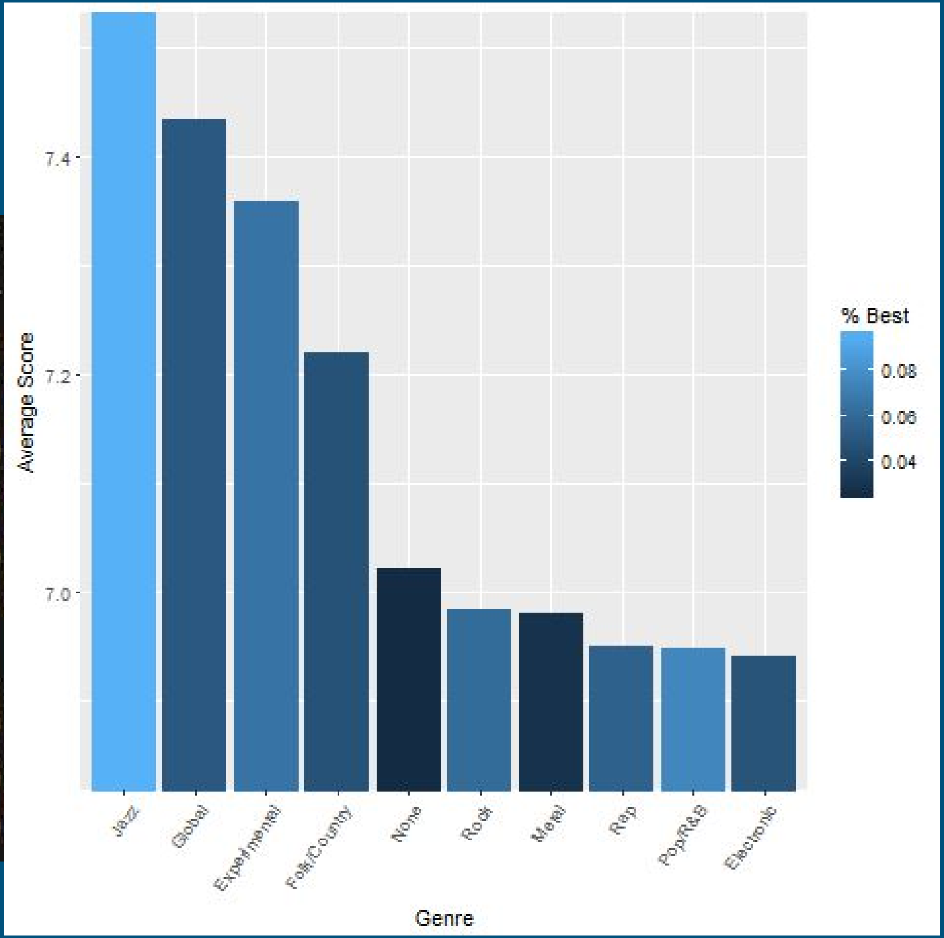
Most Popular Genre:

Rock, with 6958 Reviews (roughly 35% of the corpus)



Summary by Genre

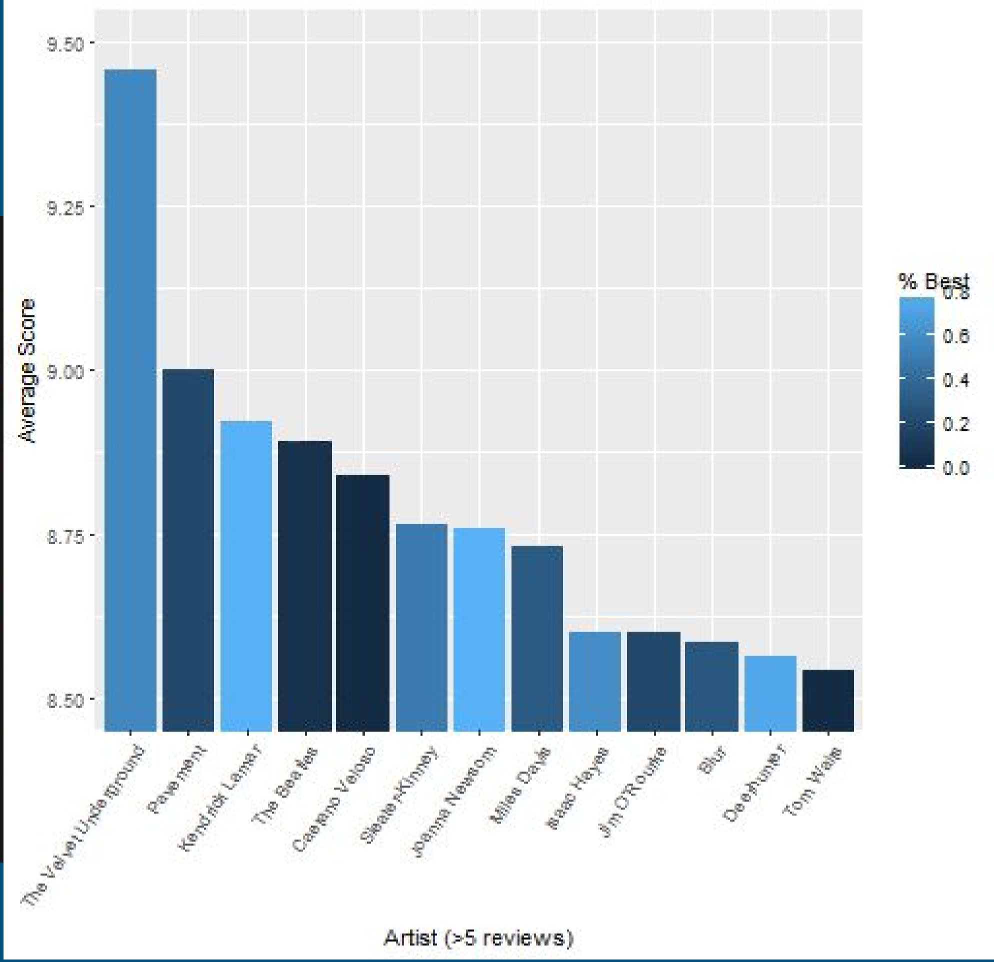
	genre	n	avg_score	std.dev	best
	<fctr>	<int>	<dbl>	<dbl>	<dbl>
1	Rock	6958	6.983616	1.336472	0.06079333
2	Electronic	4020	6.941318	1.286230	0.04800995
3	None	2324	7.020611	1.232897	0.02366609
4	Experimental	1699	7.359035	1.084305	0.06533255
5	Rap	1481	6.950304	1.259559	0.05536799
6	Pop/R&B	1157	6.948315	1.251231	0.07519447
7	Metal	781	6.980410	1.382885	0.02816901
8	Folk/Country	700	7.219857	1.039006	0.04714286
9	Jazz	257	7.568482	1.160742	0.09727626
10	Global	178	7.434831	1.020147	0.05056180



Summary by Artist

	artist	n	avg_score	std.dev	best
	<fctr>	<int>	<dbl>	<dbl>	<dbl>
1	Various Artists	718	7.290529	1.307971	0.04178273
2	Guided by Voices	25	7.232000	1.298948	0.00000000
3	David Bowie	22	7.754545	1.596045	0.13636364
4	Mogwai	21	7.266667	1.083667	0.04761905
5	The Beatles	21	8.890476	1.314117	0.04761905
6	Of Montreal	20	6.745000	1.172705	0.05000000
7	The Fall	20	7.235000	1.924434	0.05000000
8	Animal Collective	19	7.636842	1.389329	0.26315789
9	Neil Young	19	7.200000	1.452966	0.10526316
10	Robert Pollard	19	5.789474	1.795951	0.00000000

... with 800 more rows



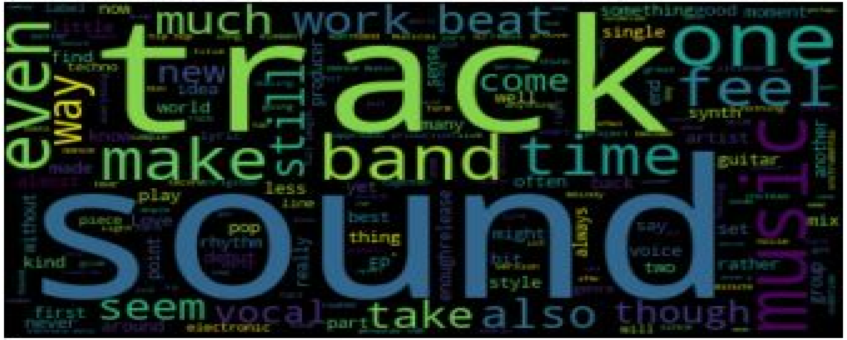
Textual Analysis Overview

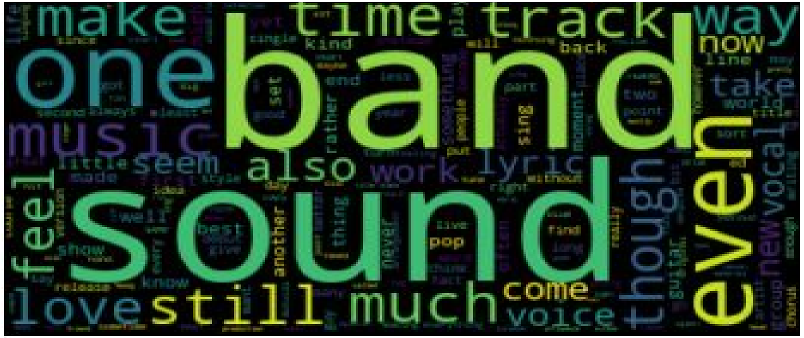
- For analyzing the text itself, we used Python’s join function to add all of the text into a single string, which is stored in one cell of the DataFrame for each review.
- Once we did this, we could easily separate the data by genre (or any other feature) then analyze just that subset of reviews.
- In practice, we ended up using POS tagging, frequency distributions, wordclouds, and TF-IDF to help us determine and visualize which words were used the most, and if there were any notable differences between genres.
- Textual analysis was conducted in Python using the NLTK package.

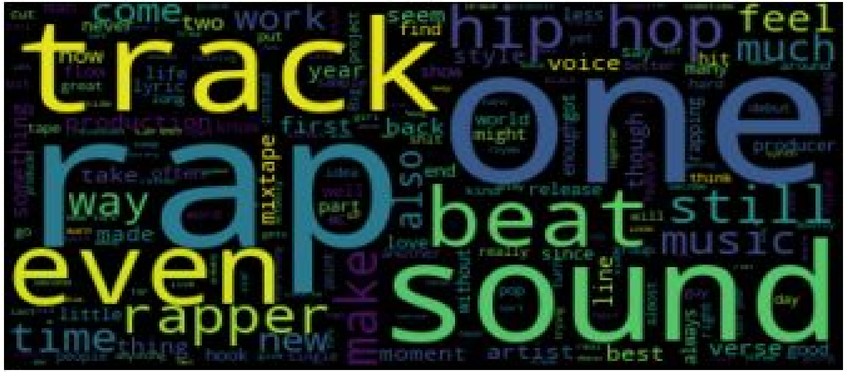
Wordcloud Methodology

- Wordclouds were chosen as an easy way to visualize token distribution over entire corpora of reviews.
- Because the words “album”, “record”, and “song” appear heavily in every genre, these words were removed from each string before tabulating the wordclouds in the hope of garnering more unique words per genre.
- Nonetheless, many other words appear frequently throughout the corpus as well, but with less frequency.
    - More work could be done to determine what words are used commonly in all genres in order to find out which words are unique to each.
- The python packages wordcloud and matplotlib were used to generate the wordclouds.

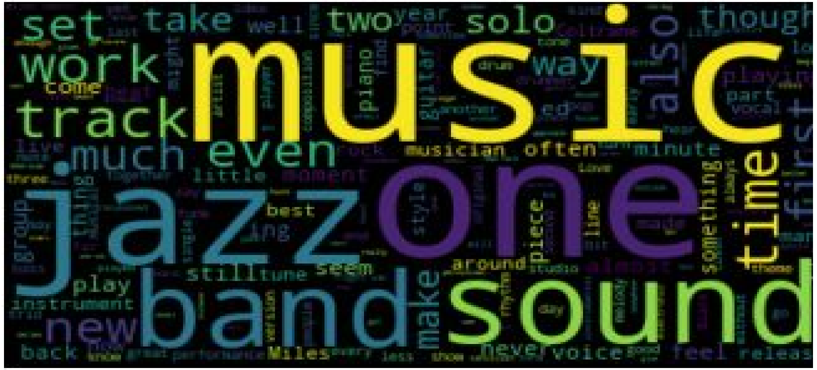
Wordclouds - All, Electronic, Rock, Rap

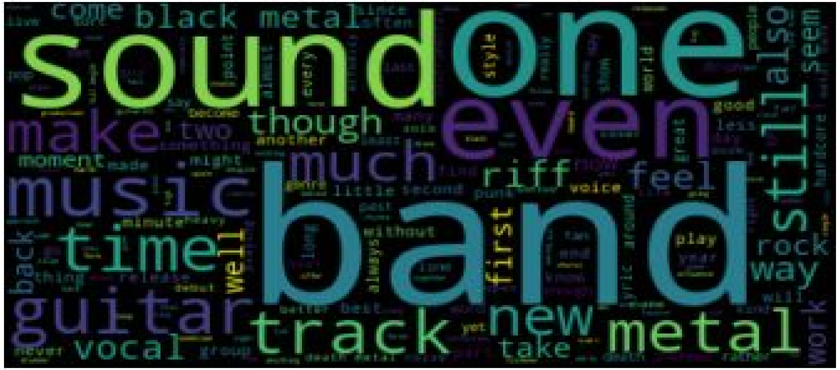


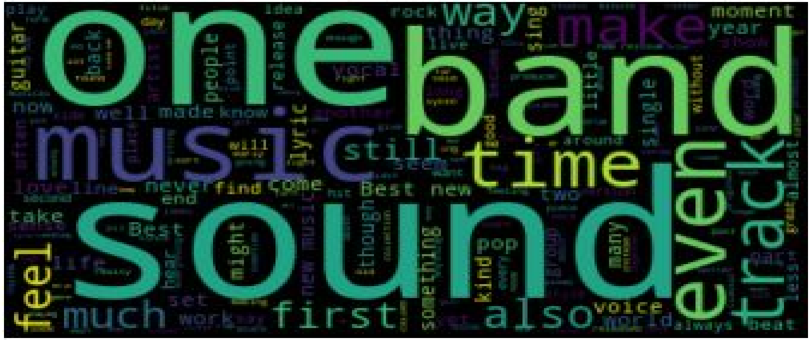


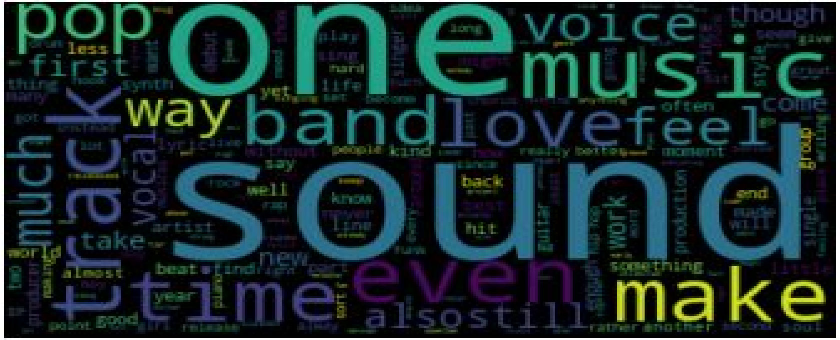


Wordclouds - Jazz, Metal, Best, Pop/R&B









Wordcloud Conclusions

- Generally, reviews of different genres exhibit many of the same highest frequency words.
    - Words like “sound” and “one” are ubiquitous, while “band” is very frequent for any genre that typically plays with a band (rock, jazz, metal, etc)
- Certain words that are more genre specific tend to appear frequently as well.
    - Words like “riff” and “guitar” for metal, “beat” for rap and electronic, “voice” for pop/R&B, and “love” for rock and pop/R&B.
- Reviews for albums deemed “best” don’t appear to have significantly different words than reviews for all albums more generally.

Popular Adjectives

- Once each word was tagged, we could use a frequency distribution over a subset of the words to see what adjectives were used most in reviewing each genre.
- Again, many common adjectives were found among reviews - "new", "first", "other", "own", and "much" appeared as the top 5 for nearly every subset we tested, including the dataset as a whole.
- However, looking at the top 25 from each we also find some representative words.
  - Metal - "black" and "heavy"
  - Pop - "vocal"
  - Rock - "hard"
- Generally, adjectives were very similar across genres.

TF-IDF Background

- Term frequency-inverse document frequency is a way of reflecting how important a word is in a document relative to its entire corpus.
  - For us, this is useful to determine words that are most important to each genre.
- It is calculated by weighing the amount of times a word appears in each document to its total frequency in the corpus, penalizing words that appear heavily in both.
- To reduce computation time, we used only the first 100 reviews from rock, metal, jazz, pop/R&B, electronic, rap, and best.
  - Even so, this still took over an hour to run...imagine how long it would take for the entire corpus!

TF-IDF Results

Top words in rock

- Word: Writing, TF-IDF: 0.00129
- Word: Guitar, TF-IDF: 0.00117
- Word: Them, TF-IDF: 0.00101
- Word: Petty, TF-IDF: 0.00089
- Word: Go, TF-IDF: 0.00076

Top words in jazz

- Word: The, TF-IDF: 0.01318
- Word: One, TF-IDF: 0.00613
- Word: Solo, TF-IDF: 0.00204
- Word: Best, TF-IDF: 0.0019
- Word: Coltrane, TF-IDF: 0.00168

TF-IDF Results Cont.

Top words in metal

- Word: Metal, TF-IDF: 0.00993
- Word: It, TF-IDF: 0.0037
- Word: And, TF-IDF: 0.00206
- Word: Danzig, TF-IDF: 0.00154
- Word: Bands, TF-IDF: 0.00139

Top words in electronic

- Word: In, TF-IDF: 0.00345
- Word: Electronic, TF-IDF: 0.00241
- Word: Little, TF-IDF: 0.00173
- Word: Work, TF-IDF: 0.00154
- Word: Track, TF-IDF: 0.00147

TF-IDF Results Cont.

Top words in best

- Word: It, TF-IDF: 0.00307
- Word: And, TF-IDF: 0.00165
- Word: Work, TF-IDF: 0.00137
- Word: Sings, TF-IDF: 0.00105
- Word: The, TF-IDF: 0.00093

Top words in rap

- Word: lil, TF-IDF: 0.00183
- Word: Mixtape, TF-IDF: 0.00179
- Word: Hip-Hop, TF-IDF: 0.00167
- Word: Rappers, TF-IDF: 0.00154
- Word: Thug, TF-IDF: 0.00147

TF-IDF Results Cont.

Top words in pop/R&B

Word: r, TF-IDF: 0.00288

Word: b, TF-IDF: 0.0014

Word: Black, TF-IDF: 0.00106

Word: Madonna, TF-IDF: 0.00095

Word: Singer, TF-IDF: 0.0008

Most Representative Words Per Genre

- Rock - Guitar, Petty
- Jazz - Solo, Coltrane
- Metal - Danzig, Bands
- Electronic - Work, Track
- Best - Work, Sings
- Rap - Mixtape, Rappers
- Pop/R&B - Singer, Madonna

TF-IDF Conclusions

- Due to our smaller corpus of reviews, named entities that were frequently referenced appeared in our results among other words.
- We were still able to identify words that intuitively made sense and seemed representative of each genre.
    - Electronic and best were the least useful, however. This agrees with our findings that “best” reviews aren’t significantly different than others.
- Along with our study of adjectives, this helped us to further determine the linguistic characteristics of our corpus as they vary by genre, one of our main goals in analysis.
- In the future, it would be illuminating to compute TF-IDF statistics across the entire corpus, even though it was not feasible for this project.

Limitations

- Memory Usage
  - Due to the computational complexity of many of the tasks involved in analysis (such as n-gram segmentation and TF-IDF statistics), memory usage became a common problem in analysis.
  - Adaptations included keeping only the variables required loaded into the current Python session and closing other windows to keep the computer's overall memory usage lower.
- Time
  - Many operations (specifically those involving TF-IDF or over the entire corpus) took minutes or hours to run.
  - Adaptations included running large operations (like scraping) in multiple chunks, or operating on a subset of the data rather than the whole thing (in the case of TF-IDF).
- Encoding
  - Encoding issues led to some strange characters when the data was imported to Python.

Conclusion

- We scraped a corpus of ~20,000 Pitchfork reviews and other identifying features, then performed exploratory and textual analysis on it.
- We were able to calculate summary statistics relating to each genre, as well as perform analysis of review text through wordclouds, frequency distributions, and TF-IDF to make conclusions about how each genre is represented in the dataset.
  - We determined some representative words that help define each genre through wordclouds and TF-IDF.
  - We found that adjective use is fairly consistent among reviews, at least on a large scale.
  - We were unable to learn much about what distinguishes review of “good” albums versus those that aren’t.

Future Work and Improvements

- A wider range of websites would provide a greater amount of information, allowing for comparison between sites (genres, artists, etc.) as well as between different reviews of the same album/artist.
- Interesting additional features to include would be reviewer name, record label, and related artists (as mentioned in the review).
- More analysis could be done on determining which words are most unique to each genre, as well as using time-series techniques to judge how scores and word usages change over time.
- Eventually, developing a model that could predict the genre (or source) of a review based on its words would be extremely cool!

That’s All, Folks!

Any Questions? Comments? Concerns?