• Over the past few decades, music has gotten louder, faster, and less acoustic. What else has changed? Are songs more repetitive? Have a different sentiment? Have a greater or smaller variety of musical qualities?

• This can be a pretty big project, requiring a bunch of data wrangling at the beginning! Choose a piece of analysis that is interesting to you.

GOALS

◦ Extract features from data for Billboard Hot 100 songs. Features can be things like EchoNest features, length of time track is on top 100 list, overall artist popularity, overall track popularity, sentiment, lexical density, popular n-grams, frequency of most popular n-grams, genre, length of title...

◦ Measure the distribution of [feature] as a function of time, and then explore patterns.

ANALYSIS IDEAS

◦ Plot the distribution (mean, variance) of each feature as a function of time: either over decades or seasonally.

▪ EchoNest has a blog post on this for their audio features, looking at trends over the decades, and other people have done word frequency analysis over time (albeit both use slightly different datasets than Top 100). If you choose this track, think about looking at **lyrics** **n-grams**or **lexical density** or **titles** or**time on the charts** or **measuring variance** or **measuring seasonal trends**instead of just plotting means of individual features over long time periods.

◦ Write a classifier that predicts **whether a current Top 100 track will still be on the Top 100** in the next n weeks. What features (other than current rank and current artist popularity) were most important in each era? Remember to use train and test sets.

**◦ Cluster hit tracks** from different eras in n-dimensional feature space. Do the clusters change with time? Does the level of homogeneity change with time?

◦ Can you find tracks from recent charts that are **outliers**? that are more **similar to hit songs from the past** than to the rest of today’s hit songs (or vice versa)?

INSTALLATIONS

• MusixMatch

◦ Get API key  <https://developer.musixmatch.com/mmplans>. Everyone should get their own key since there’s a download limit per key.

◦ Install Sphinx to generate documentation: pip install -U Sphinx

◦ Install MusixMatch - if you want to use lyrics features

▪ download tar.gz file and egg file from  <https://pypi.python.org/pypi/musixmatch/0.9>

▪ In the download directory, run python setup.py bdist\_egg

▪ In the download directory, run sudo easy\_install <name of downloaded egg file>

▪ In the download directory, run  python [setup.py](http://setup.py/) build\_sphinx

◦ Retrieve API key from your email. Add it as a variable to your .bash\_profile:

▪ in ~/.bash\_profile: export MUSIXMATCH\_APIKEY=<your api key here>

▪ in terminal: source ~./bash\_profile

• Billboard

◦ pip install billboard

• EchoNest - if you want to use audio features

◦ pip install spotipy

◦ Register @ [https://developer.spotify.com/my-applications/#!/applications](https://developer.spotify.com/my-applications/%23!/applications) to get Client ID

◦ Examples: <https://github.com/plamere/spotipy/blob/master/examples/audio_features.py>

• pandas

◦ pip install pandas

• scikit-learn (not necessary to read data, but you’ll probably want it for analysis)

◦ on a mac: pip install -U scikit-learn

• nlkt - if you want to use lyrics features

◦ pip install nlkt

GATHER DATA USING APIs (see example scripts)

**• MusixMatch - you’ll build these docs with Sphinx. There’s also a copy in the Dropbox folder.**

**• Billboard**<https://github.com/guoguo12/billboard-charts>

**• Spotify (for EchoNest features)**<http://spotipy.readthedocs.io/en/latest/>

GATHER DATA

     Decide: how do you want to store the data? On Mode (hive)? In a text file? In a python dump file? What is the primary key (unique identifier)? What columns do you want accessible? From which source do you get each one?

• Use Billboard to pull artist, track, rank? for each chart of interest. (I recommend the Top 100 charts; this is the one I use in the example script.) You’ll need to decide how far back you want to go, and how frequently you want to sample charts (every week? every month?). Keep in mind that there’s a limit to how much data you can pull using each API key, so make these decisions only once. :)

• Search MusixMatch to pull lyrics for those tracks. Should match on title and artist. Will need to check if fuzzy matching is necessary - check recall numbers! Can try using Echonest ID; there’s a possibility this is the “ID” field in the Echonest audio features dict. But it’s also highly likely that Spotify changed the IDs when they acquired Echonest.

• Search Spotify for EchoNest features of matched songs. Can use Spotify IDs returned by the Billboard API.

• Create composite matched data set containing Billboard dates and ranks plus MusixMatch lyrics and EchoNest features.

• Optional: process lyrics using nlkt. Optional preprocessing steps: Stemming, Stop Word Elimination, Tokenizing, Chorus De-Duping. Optional features to measure: Lexical Density, n-grams, Sentiment Analysis