Algorithm Plans

Wednesday, October 14, 2015 8:37 PM

- Read file in
  - File contents
    1:Name:[info]:A:A:B:…
    2:…
  - Convert to data structures - dict/array:=[id]=vector
  - Convert users to vectors ^
    - Follow weighting format - categories.yml
- Bucket
  - Figure out each bucket (see criteria)
  - Bucketize once; bucketize again
  - Data structures for storing vectors nicely
- Euclidean - build on past code
  - "heap" of vectors
  - Lowest distance
  - Repeat!
  - Return list of IDs and matcher
  - Make sure to have basic checks, ie preferences (hetero, homo, etc)
- Playlist
  - Assume arbitrary criteria
  - Return list of IDs and matches

Order to work on

1. Taking data structure in from file and print out from file
2. Hold off on bucket and playlist
3. Focus on Euclidean first, everything else second
4. Skeletons for all methods though
5. Expects
   a. Read data structure from file
   b. Print new data structure - list of matches - to file
   c. Make returns actionable
   d. Full skeleton for Euclidean
   e. Jay has skeleton for bucket

    f. Playlist - really weak skeleton
    g. Some code for measures of compatibility - gender, preference
6. One big python file should be fine
    a. 300 lines of code tops
    b. Numpyl - useful for computing distance, sorting, etc
    c. Package to deal with options parsing
    d. Standard libraries are otherwise good
    e. No ML libraries needed - dropped ML