CSE 255, Winter 2015: Homework 4

Instructions

Please submit your solution **at the beginning of the next lecture (February 2)** or outside of CSE 4102 beforehand. Please complete homework **individually**.

Download the "50,000 beer reviews" data, as well as the "facebook ego-network" data from the course webpage:
`http://jmcauley.ucsd.edu/cse255/data/beer/beer_50000.json`
`http://jmcauley.ucsd.edu/cse255/data/facebook/egonet.txt`

Code is provided on the course webpage showing how to load and perform simple processing on the data. Executing the code requires a working install of Python 2.7 with the scipy packages installed. This week's code sample uses `networkx` and `matplotlib`, though these can be ignored as they are only for visualization.

Tasks

1. From the 50,000 beer reviews data, construct features as shown in the code example from lecture 3, i.e.,
   `X = [[x['review/overall'], x['review/taste'], x['review/aroma'], x['review/appearance'], x['review/palate']] for x in data]`
   Perform k-means clustering on the data, with two centroids initialized to
   `centroids = [[0,0,0,0,1],[0,0,0,1,0]]`

   (a) What are the centroids of the two clusters after convergence, and how many of the 50,000 points are assigned to each of them (1 mark)?

   (b) What is the *sum of squared deviations* of the points from their cluster centroids (1 mark)?

   (c) Can you find centroids with better (i.e., lower) sums of squared deviations? Find a solution with two centroids and a solution with three. Report both the centroids and the sum of squared deviations for each (1 mark).

2. How many connected components are in the facebook ego-network graph, and how many nodes are in the largest connected component (1 mark)?

3. Implement *clique percolation* and apply it to the facebook ego-network graph. How many communities are discovered after running clique percolation with with 4-cliques, and which nodes are their members (1 mark)? *Hint: The code from lecture 3 includes a snippet to extract all 3- and 4-cliques from the ego-network, which may be useful:* `http://jmcauley.ucsd.edu/cse255/code/lecture3.py`